数据驱动的不确定性下概率可达集的精细调谐凸近似

Pengcheng Wu、Sonia Martinez，*IEEE*会士，Jun Chen，*IEEE*会员

arXiv:2303.01549v2 [cs.RO] 3 Feb 2024

摘要—**本文提出了一种机制，用于精细调谐不确定动态系统的概率可达集（PRS）凸近似。我们考虑不确定性无界的情形，在这种情况下，可能无法找到系统的有界可达集。因此，我们转而寻找一个能够以高置信度界定系统状态的概率可达集。我们提出的数据驱动方法基于经快速傅里叶变换（FFT）加速的核密度估计器（KDE），并针对不确定性建模进行定制，以高效地求得概率可达集。然而，概率可达集的非凸形状可能使其不适用于后续的最优设计。为此，我们构建了一个混合整数非线性规划（MINLP）问题，其解为一个最优的 n边凸多边形，用于近似概率可达集。基于这一问题构型，我们提出一种启发式算法，在确保准确性的同时高效地求得该凸集。我们通过全面的案例研究对该算法进行测试，结果表明其具有近最优性、准确性、效率和鲁棒性。本研究的优势为不确定动态系统在安全关键型实时运动规划中的应用铺平了道路。**

实践者须知——**本研究旨在实现不确定性下动态系统的安全关键实时运动规划。确保不确定动态系统安全运行的一种常用方法是可达性分析。然而，面对以下挑战时，该方法可能效果不佳：无界不确定性、未知分布、通用性、凸性和效率。为解决这些问题，我们首先提出一种数据驱动方法，用于建模任意未知不确定性，并以较高置信度获得一个包含系统状态的集合；随后，我们提出一种算法，以高效地为该集合求得紧致的凸多边形近似。这显然有利于实际运动规划。在运动规划中考虑碰撞规避时，紧致的凸近似能够扩大可行搜索区域，从而有望规划出更优的轨迹。此外，该算法的效率确保了运动规划能够实时实现。**

索引词——**概率可达集、凸近似、不确定动态系统、核密度估计器、混合整数非线性规划**

I. 引言

在城市地面和空中交通环境中部署自主系统，需要开发新型实时碰撞检测与消解算法 [1]–[3]。然而，此类系统中普遍存在的不确定性引发了对其安全性 [4]的重要担忧。这些不确定性可能由多种原因造成，例如，由于缺乏对底层机制的理解，或简单假设与复杂现实之间存在差距而产生的认知不确定性；由于系统固有的随机性而产生的偶然不确定性；或传感器测量误差 [5]–[8]。确保不确定动态系统安全运行的一种常用方法是基于可达性分析 [9]。这种基于集合的方法会计算随机动态系统在所有初始条件、容许输入和参数下可到达的状态所构成的可达集 [10]。过去几十年来，可达性分析日益受到研究人员的广泛关注 [11]。在以往的文献中，研究人员考虑不确定性的相关知识，提出了许多算法，利用详细的系统信息来确定可达集。然而，面对以下挑战，现有算法可能效果不佳，甚至失效：

1）无界性：当不确定性无界时，可能无法为不确定动态系统确定一个有界的可达集。然而，由于物理限制，大多数现实应用都需要有界的可达集。

2）未知分布：大多数文献通常假设不确定性服从高斯分布。然而，在实践中，由于系统互连未知且存在非线性，不确定性可能服从非高斯分布，甚至呈现经验分布 [12], [13]。

3) 通用性：为了计算可达集，许多算法需要预先获得详细的系统信息。然而，在某些应用中，例如只能通过实验或仿真接触到的复杂信息物理系统，这些详细信息无法获得。许多算法还依赖于对动力学或不确定性的严格假设，而这些假设往往不切实际。

4) 凸性：大多数算法计算得到的可达集是不规则的或非凸的，因此在实际解决碰撞规避问题时难以处理。

5) 效率：为了实现不确定动态系统的实时运动规划与控制，必须高效地计算可达集。然而，许多算法的计算开销很大。

为应对这些挑战，接下来我们将首先回顾一些相关文献，然后介绍本文的贡献。

Pengcheng Wu 任职于加利福尼亚大学圣迭戈分校机械与航空航天工程系（地址：加利福尼亚州拉霍亚，邮编 92093），同时任职于圣迭戈州立大学航空航天工程系（地址：加利福尼亚州圣迭戈，邮编 92182）pcwupat@ucsd.edu,pwu@sdsu.edu

Sonia Martinez 任职于加利福尼亚大学圣迭戈分校机械与航空航天工程系（地址：加利福尼亚州拉霍亚，邮编 92093）soniamd@eng.ucsd.edu

Jun Chen 任职于圣迭戈州立大学航空航天工程系，地址：加利福尼亚州圣迭戈，邮编 92182jun.chen@sdsu.edu

A. 文献综述

计算有界不确定性下可达集的一种经典方法是进行 Hamilton-Jacobi（HJ）可达性分析。该方法将问题表述为一个偏微分方程，可通过数值方法求解。尽管该方法在低维情况下具有可计算性，但在更高维度下扩展性较差 [1], [14]。此外，HJ 可达性分析需要有关动力学或不确定性的详细信息，并且依赖于诸如参数不确定性不存在时间相关性等强假设，因此在一些现实场景中无法使用 [1]。

当不确定性无界时，由于可达集也会变得无界，通常无法保证 100% 确定地实现碰撞规避。研究人员转而寻求一条碰撞概率不超过某一风险界限的轨迹。实现这一目标的一种直接方法是评估生成的候选轨迹的碰撞概率。在碰撞概率低于风险界限之前，不会采用该候选轨迹。然而，由于没有用于评估碰撞概率的闭式表达式，这种方法面临诸多困难 [15], [16]。另一种间接但有效的方法是计算概率可达集（PRS），而非精确的可达集[17], [18]。概率可达集（PRS）是系统能够以一定概率到达的所有可能状态构成的集合。通过这种方法，可以将对系统施加的、用于限制碰撞概率的随机约束转化为确定性约束，使生成的轨迹不与概率可达集（PRS）相交 [19]。该方法的优势在于，不仅适用于无界不确定性，也适用于有界不确定性。随着风险界限趋近于 0%，概率可达集（PRS）会成为可达集的超集 [17]。本文采用概率可达集（PRS）方法。

许多现有研究假设不确定性 [20], [21] 服从高斯分布。Blackmore 等人 [19] 使用由高斯分布的均值和协方差确定的凸多边形（或多胞体）作为概率可达集的近似。Wu 和 Chen 等人[4], [6] 直接使用高斯分布的等值集作为概率可达集。然而，在许多场景中，不确定性属于非高斯类型 [12]。即使如此，一些研究者仍使用高斯分布来近似非高斯分布 [22]。遗憾的是，这种方法无法保证生成轨迹的风险仍处于给定界限内 [13]。相较之下，Aoude 等人 [23] 在运动规划过程中采用粒子方法评估碰撞概率。Calafiore 和 Campi [24] 采用情景方法，对随机约束进行采样，从而得到一个凸优化问题；其解以很高的置信度满足原始随机约束。然而，上述方法通常都需要大量样本才能提供概率保证，这既不切实际，计算效率也不高。另一种方法是，Han 和 Jasour 等人[13]采用基于矩的方法来刻画非高斯概率可达集。然而，他们的研究假设矩信息是先验 已知的，但这并不现实，因为这些信息只能通过传感器测量 [25]在线学习获得。此外，概率可达集通常是不规则的，这在实践中很难处理。

这些缺点促使人们研究数据驱动的可达集分析，利用从实验或仿真中获得的数据 [25]–[27]。Devonport 等人 [11] 利用 Christoffel 函数的等值集估计可达集。Christoffel 函数根据任意未知不确定性的样本得到，无需任何详细的先验信息 先验。然而，用于估计可达集的等值集可能不规则或非凸，实际处理中难以应对。受此启发，Lew 等人[1] 采用数据样本的保守凸包来近似可达集。凸包的边数未作规定，因此后续设计问题中的约束数量可能任意。包围盒可以用固定边数包围数据样本，但其保守性更强 [28]。上述所有工作旨在给定有界不确定性的情况下估计可达集；由于缺乏不确定性量化，它们无法准确近似可达集。对于任意概率分布（包括无界分布），一种自然的可达集近似方法是使用经验概率密度函数或由数据拟合得到的概率密度函数（PDF），并将其水平集作为可达集。本文采用这一方法，并考虑核密度估计器（KDE） [29]。我们的目标是解决采用此类方法时面临的重要局限，例如 KDE 缺少闭式表达式、其水平集形状不规则，以及对水平集取凸包或包围盒会导致结果过于保守且不准确。一些研究人员考虑了具有时滞的动态系统 [30]，而本文不涉及这一问题。将所提出的方法扩展到时滞不可忽略的情形，将留待未来研究。

B. 贡献与文章结构

本文提出了一种高效的凸近似方法，用于近似不确定动态系统的概率可达集，同时确保其准确性。我们的方法由两部分组成：(i) 基于数据的不确定性量化；(ii) 用于求得凸近似的优化问题构建。

The maj或 con tributi ons o f thi s paper are the followi ：1）我们提出了一种算法，通过快速傅里叶变换（FFT）加速核密度估计（KDE），从而高效地求得概率可达集（PRS）。我们定制了基于 FFT 的核密度估计，用于估计具有任意未知不确定性的概率密度函数（PDF），因此其水平集可作为这些不确定性对应概率可达集的近似。作为一种高效的数据驱动方法，基于 FFT 的核密度估计能够在线从数据中学习，因此既不需要先验信息，也不依赖强假设；

2）我们构建了一个混合整数非线性规划（MINLP）优化问题，得到一个最优的 n 边凸多边形，用于近似概率可达集。与凸包不同，用户可以任意定制边数。该优化问题并非直接使用数据样本，而是基于核密度估计中的加权网格点建立，因此易于求解。与包围盒相比，该方法得到的结果更加紧致、准确；

3）我们开发了一种启发式算法，用于高效求解所构建的 MINLP 问题。该算法采用

加权抽样，从核密度估计中的所有网格点中选取代表性网格点，从而显著降低计算复杂度。全面的案例研究表明，该算法在为概率可达集提供凸近似的同时，兼具准确性、效率、近最优性和鲁棒性。

本文其余部分安排如下。在第 II 节中，我们正式定义概率可达集（PRS）的概念，并将问题表述为为其提供凸近似。在第 III 节中，我们使用快速傅里叶变换（FFT）加速的核密度估计（KDE）对任意概率分布进行建模，并给出一种求解概率可达集的算法。在第 IV 节中，我们构建一个混合整数非线性规划（MINLP）优化框架，其求解结果可得到概率可达集的凸多边形近似。在第 V 节中，我们开发一种启发式算法，以高效求解所构建的混合整数非线性规划。在第VI 节中，我们开展全面的案例研究，以展示所提出算法的性能。在第 VII 节中，我们总结全文并对未来研究提出建议。此外，本文部分命题的严格证明见第VIII附录。

II. 问题陈述

本节考虑一个具有不确定性的离散时间动态系统，并介绍该系统的可达集和概率可达集（PRS）的概念。随后，我们阐述本文的目标。

考虑如下形式的一般离散时间动态系统

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}\left(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{\theta}_k, \boldsymbol{w}_k\right), \tag{1}$$

其中， $\boldsymbol{x}_k \in \mathbb{R}^n$ 是系统状态， $\boldsymbol{u}_k \in \mathcal{U}_k$ 是不确定控制输入， $\boldsymbol{\theta}_k \in \Theta$ 是不确定参数， $\boldsymbol{w}_k \in \mathbb{W}$ 是不确定扰动， $\boldsymbol{x}_0 \in \mathcal{X}_0$ 是初始状态。与本文中的大多数先前研究不同，集合 $\mathcal{U}_k \subset \mathbb{R}^m, \Theta \subset \mathbb{R}^p, \mathbb{W} \subset \mathbb{R}^q$和 $\mathcal{X}_0 \subset \mathbb{R}^n$ 可以是无界的。鉴于 $\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{\theta}_k, \boldsymbol{w}_k$存在不确定性， $\boldsymbol{x}_{k+1}$是一个服从未知概率分布的随机向量。

可达集的概念形式化定义如下。

定义 1 (可达集 [31])。 在时间 T，动态系统在式（1）中的可达集 $\mathcal{X}_T$ 定义为

$$\begin{aligned}\mathcal{X}_T := \big\{ & \boldsymbol{x}_T \in \mathbb{R}^n : \boldsymbol{x}_k = \boldsymbol{f}\left(\boldsymbol{x}_{k-1}, \boldsymbol{u}_{k-1}, \boldsymbol{\theta}_{k-1}, \boldsymbol{w}_{k-1}\right), \\ & \boldsymbol{x}_0 \in \mathcal{X}_0, \boldsymbol{u}_{k-1} \in \mathcal{U}_{k-1}, \boldsymbol{\theta}_{k-1} \in \Theta, \boldsymbol{w}_{k-1} \in \mathbb{W}, \\ & k \in \{1, \ldots, T\}\big\},\end{aligned}$$

其中， $\boldsymbol{x}_0$ 是初始状态， $\mathcal{X}_0$ 是初始集。

注意，由于集合 $\mathcal{U}_{k-1}, \Theta, \mathbb{W}$和 $\mathcal{X}_0$ 不局限于有界集，因此可达集 $\mathcal{X}_T$ 可能是无界的。因此，找到一个能保证状态必然位于其中的有界集可能不可行。我们希望找到一个有界集，使状态位于该有界集中的概率大于某个置信水平。为此，我们正式引入概率可达集的概念如下。

定义 2 （概率可达集（概率可达集 [18]））。 在˜时间 k，有界集 $\mathcal{X}_k$ 被定义为一个概率可达集 （方程（1）中动态系统的概率可达集，置信水平为 α 当且仅当

$$\Pr(\boldsymbol{x}_k \in \tilde{\mathcal{X}}_k) \geq \alpha.$$

概率可达集是系统以某一概率能够到达的可能状态集合˜。请注意， $\mathcal{X}_k$ 可能不存在。当˜ ˜$\alpha = 100\%$时，若存在概率可达集 $\mathcal{X}_k$，则 $\mathcal{X}_k$ 是可达集的超集。

本文假设式（1）中的动态系统是二维的，将一般维数情形留待未来研究。本文的目标有两方面：1）提出一种数据驱动方法，对任意未知的˜不确定性进行建模，并求得该动态系统的概率可达集 $\mathcal{X}_k$ ； 2）受实现不确定系统安全关键实时运动规划的启发，我们旨在通过一个有界集来近似概率可达集˜$\mathcal{X}_k$ ，该集合满足以下条件：i）**凸性**：集合边界是一个 n 边凸多边形；ii）**效率**：该集合的计算是可处理的；iii）**准确性**：状态 $\boldsymbol{x}_k$ 位于集合内的概率接近预设置信水平 α；iv）**最优性**：在保证准确性的同时，集合面积尽可能小（不过于保守）。

III. 概率可达集辨识

在本节中，我们通过基于 FFT 的核密度估计对任意未知概率分布进行建模，并开发一种在线算法，以置信水平 α求得一个概率可达集。

A. 基于 FFT 的核密度估计

我们先简要回顾基于 FFT 的核密度估计。下面介绍核密度估计的概念。

定义3 （核密度估计器（KDE） [29]）。 设$\boldsymbol{x}_k \in \mathbb{R}^2$, $k \in \{1, \ldots, M\}$ 为从由概率密度函数f 给定的二维概率分布中抽取的 M数据样本： $\mathbb{R}^2 \mapsto \mathbb{R}$。用于近似概率密度函数 f 的核密度估计器 （KDE）ˆ $f : \mathbb{R}^2 \mapsto \mathbb{R}$定义为

$$\hat{f}(\boldsymbol{x}) = \frac{1}{M} \sum_{k=1}^{M} K(\boldsymbol{x} - \boldsymbol{x}_k),$$

其中，函数 $K : \mathbb{R}^2 \mapsto \mathbb{R}$ 是核函数，它是一个对称的二维密度函数。

核函数 K 的选择对核密度估计的准确性并不关键，因此本文采用标准的二元高斯核

$$K(\boldsymbol{x}) = (2\pi)^{-1} \det(\mathbf{H})^{-\frac{1}{2}} \exp(-\frac{1}{2} \boldsymbol{x}^\top \mathbf{H}^{-1} \boldsymbol{x}), \tag{2}$$

其中， $\mathbf{H}$ 是一个对称正定的带宽矩阵。带宽矩阵的选择对于核密度估计逼近概率密度函数的准确性至关重要。本文根据 Silverman 规则 [32]选择带宽矩阵。

考虑由 N^2 网格点 $\{(x_i, y_j) \in \mathbb{R}^2 : i, j \in \{1, \ldots, N\}\}$ 在平面 $\mathbb{R}^2$ 上等间距排列形成的网格，其边界包含所有 M 数据样本 $\boldsymbol{x}_k \in \mathbb{R}^2, k \in \{1, \ldots, M\}$。网格点网格可视为一个映射

$\boldsymbol{g}: \{(i,j): i,j \in \{1,\dots,N\}\} \mapsto \{(x_i,y_j) \in \mathbb{R}^2: i,j \in \{1,\dots,N\}\}$ 定义如下

$$(x_i, y_j) = \boldsymbol{g}(i,j),\ i,j \in \{1,\dots,N\}, \tag{3}$$

其中，i，j 是坐标为（x_i，y_j）的网格点的索引。在本文其余部分中，坐标 $\boldsymbol{g}$（i，j）简写为$\boldsymbol{g}_{ij}$。

核密度估计3中给出的3 可以在 N^2 网格点 $\boldsymbol{g}$ 上进行计算，3 从而得到

$$\hat{f}_{ij} := \hat{f}(\boldsymbol{g}_{ij}) = \frac{1}{M}\sum_{k=1}^{M} K(\boldsymbol{g}_{ij} - \boldsymbol{x}_k),\ i,j \in \{1,\dots,N\}.$$

显著加速核密度估计计算的一种方法是线性分箱。我们不再将核函数 K置于 M 数据样本上，而是将其置于由网格计数加权的 N^2 网格点 $\boldsymbol{g}$上 c_{uv}，u，$v \in \{1,\dots,N\}$。网格计数c_{uv} 是用于表示网格点附近数据样本数量的权重 $\boldsymbol{g}_{uv}$。为了获得 c_{uv}，我们可以遍历每个数据样本，并将权重分配给其相邻网格点，如图1所示。

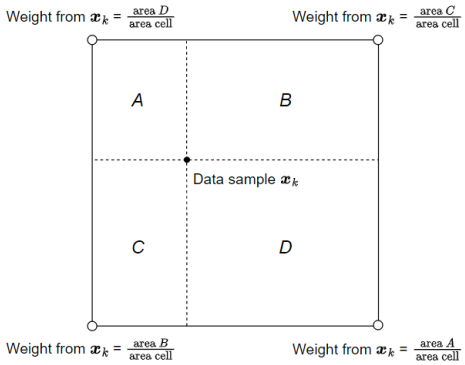


图1。二元线性分箱的图示

˜采用线性分箱策略，分箱后的核密度估计f：$\mathbb{R}^2 \mapsto \mathbb{R}$ 用于近似核密度估计ˆ f 在定义3 中表示为

$$\tilde{f}(\boldsymbol{x}) = \frac{1}{M}\sum_{u=1}^{N}\sum_{v=1}^{N} K(\boldsymbol{x} - \boldsymbol{g}_{uv})c_{uv}, \tag{4}$$

其中，c_{uv} 是由 (u,v) 索引的网格点 $\boldsymbol{g}_{uv}$ 的网格计数。

在网格 $\tilde{f}$ 上对式（4中的分箱核密度估计值进行求值，其中网格包含 N^2个网格点 $\boldsymbol{g}$，可得

$$\tilde{f}_{ij} := \tilde{f}(\boldsymbol{g}_{ij}) = \frac{1}{M}\sum_{u=1}^{N}\sum_{v=1}^{N} K(\boldsymbol{g}_{ij} - \boldsymbol{g}_{uv})c_{uv}, \tag{5}$$
$$i,j \in \{1,\dots,N\},$$

˜ ˜这表明，矩阵 $\boldsymbol{f} := [f_{ij}]$ 是 $\boldsymbol{c}:= [c_{uv}]$ 与 $\boldsymbol{k}:= [K(\boldsymbol{g}_{uv})]$的离散卷积。

离散卷积可以通过快速傅里叶变换计算，从而显著节省计算时间。令 $\boldsymbol{C}$ 和 $\boldsymbol{K}$ 分别为 $\boldsymbol{c}$ 和 $\boldsymbol{k}$的快速傅里叶变换˜， $\boldsymbol{F}$ 为 $\boldsymbol{C}$ 和 $\boldsymbol{K}$的逐元素乘积。˜ ˜那么，$\boldsymbol{f}$ 可以通过对 $\boldsymbol{F}$进行逆快速傅里叶变换提取。通过˜这种方式，我们可以得到分箱核密度估计 f_{ij}，i，$j \in \{1,\dots,N\}$ 在式（5）以实时近似任意未知概率分布的概率密度函数 f [15], [33]。

B. PRS 在线计算

针对不确定动态系统的实时运动规划，我们提出一种高效的数据驱动算法，利用基于 FFT 的核密度估计来捕获未知静态不确定性的概率可达集，其数据样本会逐步观测到。参见**算法 1**。

在算法 1的伪代码中，符号 ϵ 表示置信误差的容差。从第 10 行到第 17 行，使用快速傅里叶变换加速离散˜卷积 $\boldsymbol{c}$ 与 $\boldsymbol{k}$ 的计算，从而获得在包含 $\boldsymbol{f}$ 个网格点的网格 N^2 上求值的分箱核密度估计值 $\boldsymbol{g}$。从第 18 行到第 33 行，采用二分搜索来寻找临界分箱核密度估计值 C^{kde}，使得置信水平 α 得以达到。与 C^{kde} 对应的核密度估计的等值集，就是我们要寻找的概率可达集。

随着收集的数据样本数量增加，根据这些数据样本得到的核密度估计将收敛于未知不确定性的真实概率密度函数。然而，考虑更多数据样本需要更长的评估时间。图2 展示了数据样本数量增加对在置信水平 95% 下，通过实现算法识别网格 128^2 点上的概率可达集所需评估时间的影响。1随着数据样本数量增加，所提出的算法 1能够快速重新计算。例如，当数据样本数量处于 10^3 至 10^4, 之间时，实现算法 1 所需的评估时间始终低于 0。01s。这一结果展示了所提出的算法1 的计算能力，因此该算法能够以合理的方式处理流数据。然而，它无法保证处理随时间演变的不确定性，这将是我们未来的研究工作之一。

此外，请注意，以这种方式获得的概率可达集形状通常不规则，因此无法用于后续需要最优设计的应用。下一节将解决这一问题。

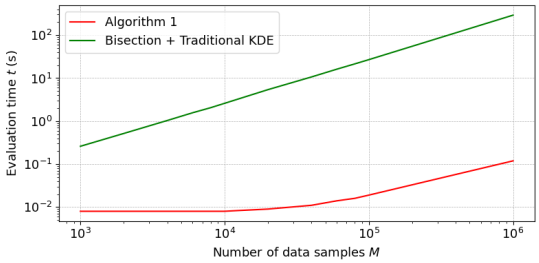


图2. 算法1的评估时间随数据样本数量增加的变化

IV. 用于概率可达集近似的混合整数非线性规划形式化

在本节中，我们将第 II 节所述的凸近似问题表述为一个混合整数非线性规划优化问题，用于处理受任意不确定性影响的二维系统的概率可达集。

算法1 概率可达集识别算法

函数 GENDS(M)
  生成 M 数据样本 $\boldsymbol{x}_k, k \in \{1, \ldots, M\}$
  返回 $\boldsymbol{x}_k$

函数 MESHGRID($\boldsymbol{x}_k$, N)
  从数据样本 $\boldsymbol{x}_k$ 中获取 $x_{\min}, x_{\max}, y_{\min}, y_{\max}$
  $\boldsymbol{h} =$ linspace($x_{\min}$, $x_{\max}$, N)
  $\boldsymbol{v} =$ linspace($y_{\min}$, $y_{\max}$, N)
  $\boldsymbol{g} =$ $\boldsymbol{h}$ 和 $\boldsymbol{v}$ 的笛卡尔积
  返回 $\boldsymbol{g}$

函数 FFTKDE($\boldsymbol{x}_k$, $\boldsymbol{g}$)
获取网格计数 $\boldsymbol{c}$，针对所有网格点 $\boldsymbol{g}$ 使用 $\boldsymbol{x}_k$ 计算核函数 $\boldsymbol{k}$ 在所有网格点 $\boldsymbol{g}$ $\boldsymbol{C} =$ 快速傅里叶变换($\boldsymbol{c}$) $\boldsymbol{K} =$ 快速傅里叶变换($\boldsymbol{k}$) $\boldsymbol{F} =$ 对以下对象进行逐元素乘积：$\boldsymbol{C}$ 和 $\boldsymbol{K}$ $\boldsymbol{f} =$ 逆快速傅里叶变换($\boldsymbol{F}$) **返回** $\boldsymbol{f}$

函数 BISECSEARCH($\tilde{\boldsymbol{f}}$, α, ϵ)
  $low =$ 最小值($\tilde{\boldsymbol{f}}$)
  $up =$ 最大值($\tilde{\boldsymbol{f}}$)
  当 $low < up$ **时执行**
    $mid = (low + up)/2$
    $\boldsymbol{z}^{\text{bin}} = (\tilde{\boldsymbol{f}} \geq mid) \cdot 1$
    $\boldsymbol{z}^{\text{mix}} =$ $\boldsymbol{z}^{\text{bin}}$ 的逐元素乘积和 $\tilde{\boldsymbol{f}}$
    $pr =$ 求和($\boldsymbol{z}^{\text{mix}}$)/求和($\tilde{\boldsymbol{f}}$)
    如果 绝对值$(pr - \alpha) \leq \epsilon$ **则**
      $C^{\text{kde}} = mid$
      返回 $\boldsymbol{z}^{\text{bin}}$, C^{kde}
    否则，如果 $pr < \alpha$ **则**
      $up = mid$
    else
      $low = mid$
  返回 $Failure$

函数 FINDPRS(M, N, α, ϵ)
$\boldsymbol{x}_k =$ GENDS(M)
$\boldsymbol{g} =$ MESHGRID($\boldsymbol{x}_k$, N)
$\boldsymbol{f} =$ FFTKDE($\boldsymbol{x}_k$, $\boldsymbol{g}$)
$\boldsymbol{z}^{\text{bin}}$, $C^{\text{kde}} =$ BISECSEARCH($\tilde{\boldsymbol{f}}$, α, ϵ)
返回 $\boldsymbol{g}$, $\tilde{\boldsymbol{f}}$, $\boldsymbol{z}^{\text{bin}}$, C^{kde}

FINDPRS(M, N, α, ϵ)

回顾一下，在通过算法 1 的实现计算公式（5）后，我们得到每个网格点 $\boldsymbol{g}_{ij}$，$i, j \in \{1, \ldots, N\}$ 分别具有分箱核密度估计值 $\tilde{f}_{ij}$，$i, j \in \{1, \ldots, N\}$。我们定义归一化权重矩阵 $\boldsymbol{w} := [w_{ij}]$ 其元素为

$$w_{ij} := \frac{\tilde{f}_{ij}}{\sum_{i=1}^{N}\sum_{j=1}^{N} \tilde{f}_{ij}}, \quad i, j \in \{1, \ldots, N\}. \tag{6}$$

混合整数非线性规划将使用这些权重而非原始数据样本来建立。这样更易于求解，因为网格点的数量远少于数据样本的数量，从而显著减少了决策变量和约束的数量。

A. 问题描述与目标函数构造

优化目标是通过确定哪些加权网格点应位于多边形内部，寻找一个边数最少的凸多边形，以高效、准确地近似概率可达集。

为此，我们引入三类决策变量来构造混合整数非线性规划优化问题：

- $2n$ continuous variables $a_k, b_k \in \mathbb{R}$, $k \in \{1, \ldots, n\}$;
- $N^2 n$ binary variables $l_{ij}^k \in \{0, 1\}$, $i, j \in \{1, \ldots, N\}$, $k \in \{1, \ldots, n\}$;
- N^2 binary variables $z_{ij} \in \{0, 1\}$, $i, j \in \{1, \ldots, N\}$.

这里，整数 $n \geq 3$ 表示凸多边形的边数；整数 N 表示每个维度上的网格点数；一对有序实数 (a_k, b_k) 表示直线 l^k 的系数，该直线是凸多边形的一条边的延长线；当且仅当网格点 $\boldsymbol{g}_{ij}$（由 (i, j) 索引）位于直线 l^k 的指定一侧时，二元变量 $l_{ij}^k = 1$ 取值为 1；当且仅当网格点 $\boldsymbol{g}_{ij}$ 位于多边形内部时，二元变量 $z_{ij} = 1$ 取值为 1。下文将详细讨论这些决策变量的作用。

我们的目标是在不违反约束的前提下，使凸多边形尽可能小。可以通过最小化位于多边形内部的网格点数量来实现这一目标。形式上，我们的优化目标函数可表示为

$$\min \sum_{i=1}^{N}\sum_{j=1}^{N} z_{ij}, \tag{7}$$

其中，$\sum_{i=1}^{N}\sum_{j=1}^{N} z_{ij}$ 是二元变量 z_{ij} 的线性函数。

B. 连续变量约束

所有不经过原点的平面直线都可表示为

$$\{ax + by - 1 = 0 : a, b \in \mathbb{R} \wedge a^2 + b^2 \neq 0\},$$

且不同的系数对 (a, b) 确定不同的直线。

一条直线 $l^k := a_k x + b_k y - 1 = 0$ 总是将整个平面分为两个半平面。我们将原点 $(0, 0)$ 所在的半平面 $\{(x, y) : a_k x + b_k y - 1 < 0\}$ 称为该直线的严格内侧，并将 $\{(x, y) : a_k x + b_k y - 1 \leq 0\}$ 称为该直线的内侧。严格内侧是同一直线内侧的真子集。

对于任意 $n \geq 3$ 条不经过原点的平面直线，由这些直线围成的区域 S 定义为

$$S := \{(x, y) : \bigwedge_{k=1}^{n} a_k x + b_k y - 1 \leq 0\}, \tag{8}$$

其中 a_k，$b_k \in \mathbb{R}$、$a_k^2 + b_k^2 \neq 0$ 和 $n \geq 3$。一方面，由于 S 是若干内侧的交集，因此 S 是一个凸集，且原点 (0, 0) $\in S$；另一方面，区域 S可能不是有界集（换言之，S 的边界可能并未形成包围）。即使有界，S 的边界也可能是边数少于 n的多边形，甚至不是多边形，而是由线段组成的图形或单独一个点。

我们要寻找能够确保 S 是一个有界的 n 边凸多边形且原点位于其内部的条件。为此，我们首先介绍以下概念。

定义 4（有向图 [34]）。有限有向图（简称有向图）是一个二元组（V，E），其中，$V \subseteq \mathbb{R}^2$ 是称为节点的有限平面点集，$E \subseteq V \times V$是由节点构成的有序对集合，称为（有向）边。有向图允许存在自环。对于 u，$v \in V$，如果有序对（u，v）$\in E$，则表示一条从 u到 v的边。

定义 5（路径 [34]）。由有限序列的平面点 $S_g := (S_g[0], \ldots\ldots, S_g[m])$ 给出的有限有向路径（简称路径），其中 $S_g[i] \in \mathbb{R}^2$，$i \in \{0, \ldots, m\}$，是一个有向图，其节点集为 $V = \{S_g[i] : i \in \{0, \ldots, m\}\}$，边集为 $E = \{(S_g[i], S_g[i+1]) : i \in \{0, \ldots, m-1\}\}$。在路径 S_g中，当且仅当存在一条边（$S_g[i]$，$S_g[j]$）时，一个节点 $S_g[i]$ 紧邻于另一个节点$S_g[j]$ 之前；两个节点是连续的，当且仅当其中一个紧邻于另一个之前；三个节点是连续的，当且仅当第一个紧邻于第二个之前且第二个紧邻于第三个之前。

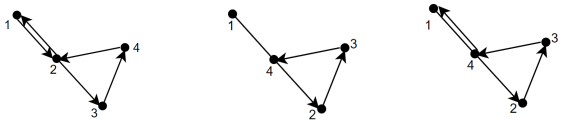
注意，如果 $S_g[i] = S_g[j]$，$i \neq j$，那么根据公理化集合论中的外延公理 [35]，$\{S_g[i], S_g[j]\} = \{S_g[i]\}$。对于 S_g 中的节点 $S_g[i]$，紧邻其前面的节点（分别是紧邻其前面的节点 $S_g[i]$）可能不存在；即使存在，也可能不唯一。如果存在一条边（$S_g[i]$，$S_g[i]$），那么 $S_g[i]$ 紧邻其自身之前。如果（$S_g[i]$，$S_g[j]$）和（$S_g[j]$，$S_g[i]$）都是 S_g 中的边，那么 $S_g[i]$ 紧邻在 $S_g[j]$ 之前且 $S_g[j]$ 紧邻在 $S_g[i]$ 之前，这种情况也可能同时发生。

定义6（有向图 [34] 的图与无向版本）。有限无向图（简称图）是一个二元组（V，E），其中 $V \subseteq \mathbb{R}^2$ 是由称为节点的有限个平面点组成的集合，而 E 是由称为（无向）边的节点无序对组成的集合。图中不允许有自环。对于 u，$v \in V$，$u \neq v$，如果集合 $\{u, v\} \in E$，则表示 u 与 v 之间有一条边。当且仅当对于 u，$v \in V$，$u \neq v$，有（u，v）$\in E' \vee$（v，u）$\in E' \iff \{u, v\} \in E$，图（$V$，$E$）是有向图（$V$，$E'$）的无向版本。

在本文中，这些定义强调平面几何性质。例如，节点的位置用二维坐标表示；连接两个不同节点的边为直线段，而自环则是一个点。取无向版本的操作会保留几何性质。例如，在有向图中共线的任意三个节点，在该有向图的无向版本中也必须共线。

定义7 (包围路径)。当且仅当满足以下条件时，路径 S_g 称为包围的：1）存在 $n \geq 3$ 个不同的节点；2）序列 S_g 中没有节点出现多次，但首尾节点相同的情况除外；3）对于连接两个节点的任意一条边，都没有其他节点位于该边上。

例如，图3中的所有路径都不是包围的。图3a中的路径 (1, 2, 3, 4, 2, 1) 违反了定义7中的条件2)，因为节点2出现了不止一次；图3b中的路径 (1, 2, 3, 4) 也违反了条件2)，因为首节点1和末节点4并不相同；图3c中的路径 (1, 2, 3, 4, 1) 满足前两个条件，但违反了条件3)，因为节点4位于边 (1, 2) 上。



(a) 违反条件2)的路径　(b) 违反条件2)的路径　(c) 违反条件3)

图3. 三种非包围路径的情形

通过包围路径的定义，我们可以引入包围图的定义。

定义8 (包围图)。图 G是包围的，当且仅当存在一条包围路径 S_g，其无向版本为 G。

例如，图 G 图 4 不是包围图，因为任何无向版本为 G的路径都不是包围路径。

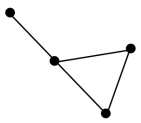


图4. 一个非包围图

显然，如果路径 S_g 是包围的，那么它的无向版本 G 也是包围的。反之，如果一条路径不是包围的，其无向版本仍可能是包围的。

受定义 7 启发，但与之略有不同，我们定义如下概念。

定义 9（形式包围符号序列）。设 S_s 为一个有限符号序列。当且仅当满足以下条件时，该序列 S_s 是形式包围的：1）存在 $n \geq 3$ 个形式不同的符号；2）除首尾符号形式相同的情况外，没有符号出现多次。

在有限符号序列 S_s 中，当且仅当存在第一个符号的一次出现$S_s[i]$ 和第二个符号的一次出现 $S_s[j]$，使得 $j = i + 1$ 时，第一个符号才紧接在 另一个符号之前；两个符号相邻当且仅当其中一个紧接在另一个之前；三个符号

相邻当且仅当第一个符号紧接在第二个之前，且第二个符号紧接在第三个之前。

有限符号序列只是一种有序结构，既不必是包围的，也不具有几何意义。为了解决这一问题，我们引入同构映射，定义如下。

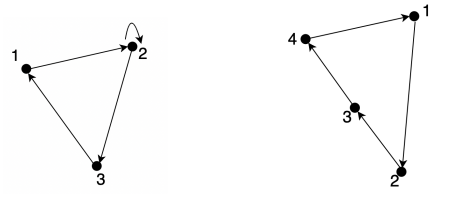
定义 10（A型同构）。设 f_{tag} 是从有限符号序列 S_s 到路径 S_g的映射。当且仅当以下性质对 f_{tag}成立时，映射 f_{tag} 是A型同构：1）单射： S_s中形式不同的符号对应于 S_g中不同的节点；2）满射： S_g中的每个节点都是 S_s中某个符号的像；3）保序：对于 S_s中的任意两个符号 $S_s[i]$、 $S_s[j]$ ，符号 $S_s[i]$ 紧接在 $S_s[j]$ 之前，当且仅当（$f_{\text{tag}}(S_s[i])$ $f_{\text{tag}}(S_s[j])$）是 S_g中的一条边；4）不在边上：对于任意两个连续符号 $S_s[i]$、 $S_s[i+1]$ 以及 S_s中的任意第三个符号 $S_s[k]$ ，若 $S_s[k]$ 在形式上不同于 $S_s[i]$ 和 $S_s[i+1]$，则 $f_{\text{tag}}(S_s[k])$不在 S_g中的边（$f_{\text{tag}}(S_s[i])$ $f_{\text{tag}}(S_s[i+1])$）上。

注释1。注意，如果存在一个从有限符号序列 S_s 到路径 S_g 的 A型同构 f_{tag} ，那么 S_s 是形式包围的，当且仅当 S_g 是包围的。我们已经看到，如果 S_g 是包围的，那么它的无向版本 G 也包围的。因此，我们得出结论：如果 f_{tag} 是从 S_s 到 S_g 的 A型同构，且 S_s 是形式包围的，那么图 G 作为 S_g 的无向版本，也包围的。

除了上述关于包围路径和图的讨论外，我们还想探讨在什么条件下能够确保路径和图非退化。

定义11（非退化路径与图）。设 S_g（分别为 G）是一个路径（分别为一个图），其节点由实数集 $\mathbb{R}^2$ 中的一组 $n \geq 3$ 个点给出。当且仅当序列 S_g 中任意三个连续节点不共线时，路径 S_g 才是非退化路径。当且仅当存在一个无向版本为 G 的非退化路径 S_g 时，图 G 才是非退化图。当且仅当路径 S_g （分别为图 G）不是非退化的，它才是退化。

要求节点数为 $n = 4$，下图5给出了两条退化路径。图5a中的路径 (1, 2, 2, 3, 1) 是退化的，因为该路径中的节点数比 $n = 4$ 少三个，而且有三个连续节点 (1, 2, 2) 共线。图5b中的路径 (1,2, 3, 4, 1) 也是退化的，因为有三个连续节点 (2, 3, 4) 共线。



（a）一条退化路径 $(1, 2, 2, 3, 1)$ with 三个节点　　（b）一条退化路径 $(1, 2, 3, 4, 1)$ with 四个节点

图5. 退化路径的两种情况

要求节点数为 $n = 4$，下图 6 给出了两个退化图，它们分别是图 5中所示两条路径的无向版本。



(a) 一个退化图 包含三个节点　　(b) A 退化 graph 包含四个节点

图6. 退化图的两种情况

请注意，非退化图（分别为路径）与包围图（分别为路径）是两个彼此独立的概念。以下图 7 给出了示例，以说明这两个概念之间的区别。

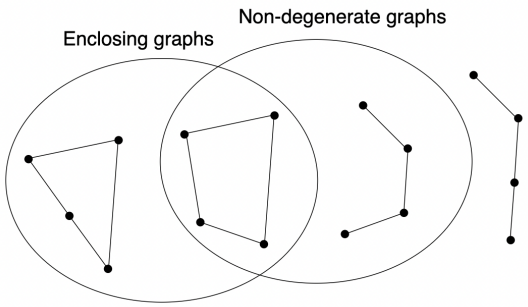


图 7. 包围图或非退化图示例　　s

定义12 (B型同构)。 设 f_{tag} 是一个从有限符号序列 S_s 到路径 S_g的映射。当且仅当以下性质对 f_{tag}成立时，映射 f_{tag} 为*B*型同构：**1) 单射**： S_s中形式不同的符号对应于 S_g中的不同节点；**2) 满射**： S_g中的每个节点都是 S_s 中某个符号的像；**3) 保序**：对于 S_s中的任意两个符号 $S_s[i]$、 $S_s[j]$ ，符号 $S_s[i]$ 紧接在 $S_s[j]$ 之前，当且仅当 ($f_{\text{tag}}(S_s[i])$ $f_{\text{tag}}(S_s[j])$) 是 S_g中的一条边；**4) 非共线性**： S_s 中任意三个连续符号的像在 S_g中不共线。

备注 2。 注意，如果存在一个从有限符号序列 S_s 到路径 S_g 的同构类型 Bf_{tag} ，并且 S_s 中有 $n \geq 3$ 个形式不同的符号，那么 S_g 及其无向版本 G 都是非退化的。

请注意，A型同构的定义10中的条件 4)与B型同构的定义12中的条件 4)不同。例如，图 8a所示的标记给出了一个从符号序列（“a”、“b”、“c”、“d”、“b”）到路径（a, b, c, d, b）的映射 f_{tag} 。映射f_{tag} 是A型同构而不是B型同构，因为存在三个连续符号“a”、“b”、“c”，其像 a、 b、 c 在该路径中共线。这违反了定义12中的条件 4)。相比之下，

图 8b所示的标记给出了另一个从符号序列（“a”、“b”、“c”、“d”）到路径（a,b,c,d）的映射 f'_{tag}。映射 f'_{tag}是B型同构而不是A型同构，因为尽管符号“d”在形式上不同于两个连续符号“a”、“b”，但节点 d 位于路径中的边（a,b）上。这违反了定义10中的条件 4)。



(a) 一种同构A型　(b) 一种同构B型

图 8. A型或B型同构的示例

我们回过头来寻找能够确保 S 在式（8）是一个有界的 n 边凸多边形，且原点位于其内部。

对于任意两条直线 $l^i := a_i x + b_i y - 1 = 0$ 和 $l^j := a_j x + b_j y - 1 = 0$，若 $D_{ij} := a_i b_j - b_i a_j \neq 0$，则这两条直线存在唯一的交点，即

$$V_{ij} := l^i \cap l^j = \left(-\frac{b_i - b_j}{D_{ij}}, \frac{a_i - a_j}{D_{ij}} \right).$$

注意 $V_{ij} = V_{ji}$。事实 $D_{ij} \neq 0$ 意味着 $a_i^2 + b_i^2 \neq 0 \wedge a_j^2 + b_j^2 \neq 0$，也意味着 $a_i \neq a_j \vee b_i \neq b_j$，即 $V_{ij} \neq$ (0,0)。这是合理的，因为这两条直线都不经过原点，因此原点肯定不是交点。D_{ij} 的符号取决于 i 和 j的顺序，即 $D_{ij} = -D_{ji}$。对于共享唯一交点的两条直线 l^i, l^j，如果我们能以原点为中心，将 l^i的法向量（a_i, b_i）逆时针旋转角度 $\theta \in (0, \pi)$，使其转至 l^j的法向量（a_j, b_j），那么 $D_{ij} > 0$; 否则，$D_{ij} < 0$。当 $D_{ij} < 0$时，我们可以交换（a_i, b_i）与（a_j, b_j）的值，使得 $D_{ij} > 0$。因此，不失一般性，我们可以假设 $D_{ij} > 0$。

为简化符号，我们定义一元运算 $i^{\oplus}$，用于 $i \in \{1, \ldots, n\}$

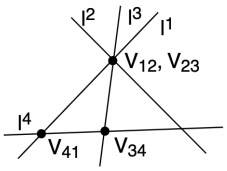
$$i^{\oplus} := \begin{cases} i+1, & i \in \{1, \ldots, (n-1)\}, \\ 1, & i = n, \end{cases}$$

其中 $i+1$ 是通常的加法运算。反过来，我们可以定义另一个一元运算 $i^{\ominus}$，用于 $i \in \{1, \ldots, n\}$，即当且仅当 $j^{\oplus} = i$ 时，$j = i^{\ominus}$。

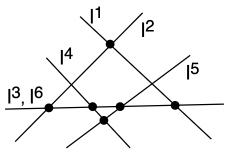
考虑 $n \geq 3$ 不经过原点的平面直线，记为 l^k, $k \in \{1, \ldots, n\}$。我们希望按照式（8）所示方式由这些直线生成的区域 S，n 是一个以原点为内部点的包围的 n 边凸多边形。实现这一点的必要条件是，至少存在 n个不同的直线交点，作为多边形的 n 节点。不失一般性，我们要求存在这样的V_{ij}形式上不同的交点 $i \in \{1, \ldots, n\}$, $j = i^{\oplus}$（这也隐含地要求$l^i \neq l^j$，$i \in \{1, \ldots, n\}$，$j = i^{\oplus}$）为此，我们构造以下约束

$$D_{ij} := a_i b_j - b_i a_j > 0, \quad i \in \{1, \ldots, n\},\ j = i^{\oplus}. \quad (9)$$

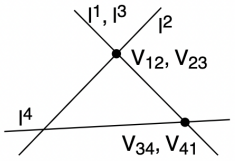
然而，满足式（9）并不能确保确实存在 n 个不同的交点（相应地，也不能确保存在不同的直线），因为交点（相应地，直线）的两种形式上不同的表示可能指向同一个交点（相应地，同一条直线），如图9所示。



(a) 点 V_{12} 和 V_{23} 是相同的



(b) 直线 l^3 和 l^6 是相同的的

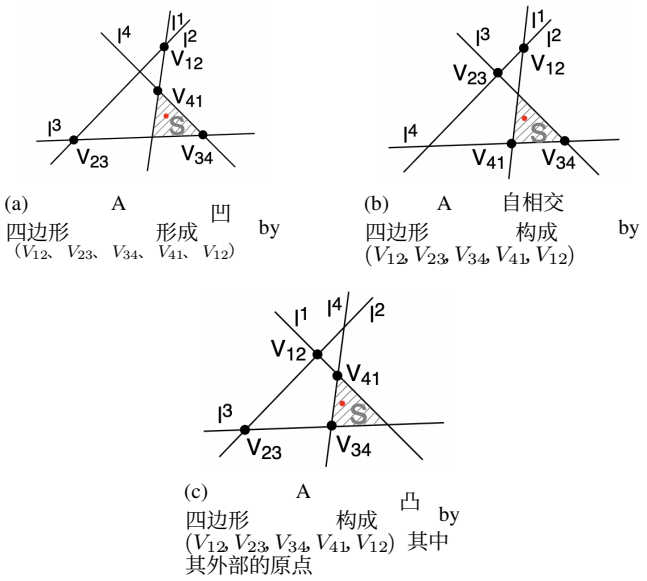


（c）直线 l^1 以及 l^3，点 V_{12} 和 V_{23}，以及点 V_{34} 和 V_{41} 相同

图9. 满足方程（9）时相同交点或直线的情形

由有限条直线 S 生成的区域的边界可以定义为路径 S_g的无向版本 G。在本文余下部分中，当图 G 是路径 S_g 的无向版本时，我们称图 G 由 S_g形成。对于任意 $n \geq 3$ 不经过原点的平面直线 l^k、$k \in \{1, \ldots, n\}$，如果存在 n 形式上不同的交点 V_{ij}、$i \in \{1, \ldots, n\}$、$j = i^{\oplus}$，我们可以通过交点 S_g的有限序列=构造一条路径：(V_{12}、V_{23}、…、$V_{(n-1)n}$、V_{n1}、V_{12})。如果由 S_g 形成的图 G是一个包围的 n 边凸多边形，且原点位于其内部，那么区域 S 的边界就与图 G相同，因此，区域 S 的边界也是一个包围的 n 边凸多边形，且原点位于其内部。因此，要使区域 S 的边界成为一个包围的 n 边凸多边形，且原点位于其内部，我们只需确保由S_g 形成的图 G是一个包围的 n 边凸多边形，且原点位于其内部。

然而，对于某些不经过原点且具有 n 形式上不同的交点V_{ij}的 $n \geq 3$ 平面直线 $i \in \{1, \ldots, n\} j = i^{\oplus}$，由 S_g构成的图 G可能并非以原点为内部点的包围 n 边凸多边形。具体而言：1）图 G 可能并不具有包围性。例如，图9c中的图 G 是一条线段，不具有包围性；2）图 G 可能退化为边数少于 n的多边形，甚至不是多边形，而是线段或一个点。例如，图9a中的图 G 是一个三角形，而非要求的四边形；3）图 G可能不是凸的（可能是凹的或自相交的），如图10a和图10b所示；4）原点 (0,0) 可能不在图 G的内部，如图10c所示。



(a) A 凹四边形 by 形成 $(V_{12}, V_{23}, V_{34}, V_{41}, V_{12})$

(b) A 自相交四边形 by 构成 $(V_{12}, V_{23}, V_{34}, V_{41}, V_{12})$

(c) A 凸四边形 by 构成 $(V_{12}, V_{23}, V_{34}, V_{41}, V_{12})$ 其中其外部的原点

图10. 不包围原点位于其内部的凸多边形的图形情况 n （阴影区域为 S 式（8）所定义的区域，红点为原点）

如上所述，约束方程（9）不足以保证由图 G 由 S_g 构成的n 边凸多边形以原点为其内部点且具有包围性。为此，我们在定理1中给出额外的约束。

在继续讨论定理 1之前，我们先引入引理 1。该引理的作用在于：一旦满足约束式（9）和式（10），我们就不再需要显式地强制所有 n 交点（分别地，直线）彼此不同。

引理1。假设有 $n \geq 3$ 条不经过原点$(0,\ 0)$的平面直线，记为 $l^k: = a_k x + b_k y - 1 = 0$、$k \in \{1, \ldots, n\}$。若有 n 个交点$V_{ij}: = l^i \cap l^j$、$i \in \{1, \ldots, n\}$、$j = i^{\oplus}$，且这些直线的系数满足以下不等式

$$\begin{aligned} &-a_k (b_i - b_j) + b_k (a_i - a_j) - (a_i b_j - b_i a_j) < 0, \\ &i \in \{1, \ldots, n\}, j = i^{\oplus}, k \in \{1, \ldots, n\} - \{i, j\}, \end{aligned} \tag{10}$$

那么，任何两种形式上不同的交点（分别为直线）的表示都指代不同的交点（分别为直线），因此所有 n 交点（分别为直线）彼此不同。

证明。参见第VIII-A节。 ■

接下来，以下定理1 给出了足以确保区域 S 在式（8）中所表示的边界是一个包围 n 边凸多边形，且原点位于其内部的条件。直观而言，约束式（10会检查每个交点V_{ij}，$i \in \{1, \ldots, n\}$，$j = i^{\oplus}$ 是否满足其所有对应的相对（$n-2$）直线 l^k，$k \in \{1, \ldots, n\} - \{i, j\}$。

定理 1。假设存在 $n \geq 3$ 条不经过原点（0，0）的平面直线，记为 $l^k: = a_k x + b_k y - 1 = 0$，$k \in \{1, \ldots, n\}$。如果存在 n 个交点$V_{ij}: = l^i \cap l^j$，$i \in \{1, \ldots, n\}$，$j = i^{\oplus}$，且这些直线的系数满足公式（10），则区域的边界 ${}_nS: = \{(x,\ y): $ 由这些直线生成的图 是由交点序列：$= (V_{12},\ V_{23},\ \ldots\ldots,\ V_{(n-1)n},\ V_{n1},\ V_{12})$ 构成的图，它定义了一个 n 边凸多边形，且原点位于其内部。

证明。参见第VIII-B节。 ■

到目前为止，我们只是推导出了约束式（9）和式（10），以确保所关注的多边形包含原点（0，0）。以下结果将这些条件的适用范围扩展至包含任意点。

推论 *1*。假设存在 $n \geq 3$ 条不经过点 $(\bar{x}, \bar{y})$ 的平面直线，记为 $l^k := a_k(x - \bar{x}) + b_k(y - \bar{y}) - 1 = 0$，$k \in \{1, \ldots, n\}$。若约束式（*9*）和式（*10*）成立，则由这些$_{k=1}$直线生成的区域 $S := {}_n\{(x, y): \bigwedge a_k(x - \bar{x}) + b_k(y - \bar{y}) - 1 \leq 0\}$ 的边界是图 G，该图由交点序列 $S_g := (V_{12}, V_{23}, \ldots, V_{(n-1)n}, V_{n1}, V_{12})$ 构成，并定义了一个包围点 $(\bar{x}, \bar{y})$ 的 n 边凸多边形。

证明。 见第 VIII-C 节。 ■

最后我们得出结论：根据推论 1，式（9）和式（10）是我们为 $2n$连续变量 $a_k, b_k \in \mathbb{R}$，$k \in \{1, \ldots, n\}$制定的约束。对于任意$n \geq 3$ 不经过任意点的平面直线，若满足式（9）和式（10），则可以保证由这些直线$- 1 \leq 0\}$ 生成的区域$_n$边界 $S := \{(x, y): \bigwedge a_k(x - \bar{x}) + b_k(y - {}_{k=1}\bar{y})$是一个包围的 n 边凸多边形，且该点位于其内部。

C. 连续变量与离散变量之间的约束

回顾一下，网格点网格是一种映射 $\boldsymbol{g}$，由 $(x_i, y_j) = \boldsymbol{g}_{ij}$，$i, j \in \{1, \ldots, N\}$ 给出，如式（3）所定义；并且我们得到了一个归一化权重矩阵 $\boldsymbol{w} := [w_{ij}]$，使得式（6）中每个网格点$\boldsymbol{g}_{ij}$，$i, j \in \{1, \ldots, N\}$ 分别具有归一化权重 w_{ij}。归一化权重最大的网格点的索引记为$\bar{\ }\bar{\ }(i, j) := \arg\max_{i,j}(\boldsymbol{w})$，其坐标为 $(x_{\bar{i}}, y_{\bar{j}}) := \boldsymbol{g}_{\bar{i}\bar{j}}$，在第 IV-B 节中起到 $(\bar{x}, \bar{y})$ 的作用。

定义以下二元变量 $z_{ij} \in \{0, 1\}$，使得 $z_{ij} = 1$ 当且仅当网格点$\boldsymbol{g}_{ij}$ 位于第IV-B节所得的多边形内。换句话说，

$$z_{ij} = 1 \implies \bigwedge_{k=1}^{n} a_k(x_i - x_{\bar{i}}) + b_k(y_j - y_{\bar{j}}) - 1 \leq 0, \tag{11}$$
$$i, j \in \{1, \ldots, N\},$$

$$z_{ij} = 0 \implies \bigvee_{k=1}^{n} a_k(x_i - x_{\bar{i}}) + b_k(y_j - y_{\bar{j}}) - 1 > 0, \tag{12}$$
$$i, j \in \{1, \ldots, N\}.$$

为了将上述逻辑约束转换为等价的代数约束，我们引入新的二元变量

$l_{ij}^k \in \{0,1\}$ 使得当且仅当网格点 $\boldsymbol{g}_{ij}$ 位于直线 l^k的内侧时，$l_{ij}^k = 1$ 成立。也就是说，

$$l_{ij}^k = 1 \implies a_k(x_i - x_{\bar{i}}) + b_k(y_j - y_{\bar{j}}) - 1 \le 0, \quad i,j \in \{1,\dots,N\}, \quad k \in \{1,\dots,n\},$$

$$l_{ij}^k = 0 \implies a_k(x_i - x_{\bar{i}}) + b_k(y_j - y_{\bar{j}}) - 1 > 0, \quad i,j \in \{1,\dots,N\}, \quad k \in \{1,\dots,n\},$$

其中 $n \in \{x \in \mathbb{N}: \ x \ge 3\}$。这些可以通过大-$M$ 表示法 [36] 表述为

$$a_k(x_i - x_{\bar{i}}) + b_k(y_j - y_{\bar{j}}) - 1 \le M_{ijk}^1(1 - l_{ij}^k) \quad (13)$$
$$i,j \in \{1,\dots,N\}, \quad k \in \{1,\dots,n\},$$

$$-a_k(x_i - x_{\bar{i}}) - b_k(y_j - y_{\bar{j}}) + 1 < M_{ijk}^2 l_{ij}^k \quad (14)$$
$$i,j \in \{1,\dots,N\}, \quad k \in \{1,\dots,n\},$$

其中，参数 M_{ijk}^1 M_{ijk}^2 i $j \in \{1,\dots,N\}$ $k \in$、$\{1,\dots,n\}$、、、均为足够大的常数。通常会针对不同约束分别设定这些常数，但在本文中，我们只需为它们选取一个共同的常数。

z_{ij} 与 l_{ij}^k 之间的逻辑关系为

$$z_{ij} = 1 \implies \sum_{k=1}^{n} l_{ij}^k = n \quad i,j \in \{1,\dots,N\},$$

$$z_{ij} = 0 \implies \sum_{k=1}^{n} l_{ij}^k \le (n-1) \quad i,j \in \{1,\dots,N\},$$

这些可以表述为代数约束

$$\sum_{k=1}^{n} l_{ij}^k \ge n z_{ij} \quad i,j \in \{1,\dots,N\}, \quad (15)$$

$$\sum_{k=1}^{n} l_{ij}^k \le (n-1) + z_{ij} \quad i,j \in \{1,\dots,N\}, \quad (16)$$

其中 $n \in \{x \in \mathbb{N}: \ x \ge 3\}$。

这样，我们便将原始逻辑约束式（11）和式（12）转化为等价的代数不等式式（13）、式（14）、式（15）和式（16）。这些约束给出了 $2n$ 连续变量a_k、b_k $\in$ $\mathbb{R}$、k $\in$ $\{1,\dots,n\}$与 N^2n二元变量$l_{ij}^k \in \{0,1\}$、 i、$j \in \{1,\dots,N\}$、 $k \in \{1,\dots,n\}$、 N^2二元变量 $z_{ij} \in \{0,1\}$、i、 $j \in \{1,\dots,N\}$之间的关系。

D. 离散变量的约束

我们希望确保网格点 $\boldsymbol{g}_{\bar{i}\bar{j}} := (x_{\bar{i}}, y_{\bar{j}})$位于第 IV-B 节所得的多边形内。回顾一下，当且仅当网格点 $\boldsymbol{g}_{ij}$位于多边形内时，二元变量 $z_{ij} = 1$ 取值为真。因此，我们施加以下约束：

$$z_{\bar{i}\bar{j}} = 1. \quad (17)$$

事实上，式（17）并非必要，因为 $(x_{\bar{i}},\ y_{\bar{j}}) \in S := \{(x, y): \bigwedge_{n} a_k(x - x_{\bar{i}}) + b_k(y - y_{\bar{j}}) - 1 \le 0\}$ 由第 1 节$_{k=1}$IV-B中的推论给出，这意味着该网格点必须位于多边形内部。不过，为了表达得更清楚，我们在此将其明确列出。

该凸多边形旨在近似概率可达集。为确保概率可达集达到一定的置信水平，我们要求位于多边形内部的网格点的归一化权重之和 w_{ij} 大于该置信水平 α。也就是说，

$$\sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij} z_{ij} \ge \alpha. \quad (18)$$

式 (17) 和式 (18) 导出两个约束，我们针对 N^2 二元变量 $z_{ij} \in \{0,1\}$、 i、 $j \in \{1,\dots,N\}$ 进行形式化。

E. Formulation of MINLP Optimization Framework

为求解第 IV-A 节所述的问题，我们$_N$ $_N$选择了式 (min $\sum\sum z_{ij}$ 中的目标函数 7)，并$_{i=1\ j=1}$引入了以下三类决策变量，其$n \in \{x \in \mathbb{N}: \ x \ge 3\}$ 见第 IV-A 节：

- $2n$ continuous variables $a_k, b_k \in \mathbb{R}$, $k \in \{1,\dots,n\}$;
- $N^2 n$ binary variables $l_{ij}^k \in \{0,1\}$, $i,j \in \{1,\dots,N\}$, $k \in \{1,\dots,n\}$;
- N^2 binary variables $z_{ij} \in \{0,1\}$, $i,j \in \{1,\dots,N\}$.

汇总前述约束：第 9 式和第 10 式（第 IV-B 节）；第 13 式、第 14 式、第 15 式和第 16 式（第 IV-C 节）；第 17 式和第 18 式（第 IV-D 节）。

我们将第II 节中所述的问题正式表述为如下混合整数非线性规划优化框架。所有坐标均在全局坐标系中表示。若未特别说明，$\forall i,\ j \in \{1,\dots,N\}$, $\forall k \in \{1,\dots,n\}$。

$$
\begin{aligned}
&\min \sum_{i=1}^{N}\sum_{j=1}^{N} z_{ij} \\
&\text{s.t.} \quad a_i b_j - b_i a_j > 0, \quad \forall i \in \{1,\dots,n\}, j = i^{\oplus}; \\
&\quad -a_k(b_i - b_j) + b_k(a_i - a_j) < a_i b_j - b_i a_j, \quad \forall i \in \{1,\dots,n\},\ j = i^{\oplus},\ \forall k \in \{1,\dots,n\} - \{i,j\}; \\
&\quad a_k(x_i - x_{\bar{i}}) + b_k(y_j - y_{\bar{j}}) - 1 \le M^1_{ijk}(1 - l^k_{ij}), \quad \forall i,j,k; \\
&\quad -a_k(x_i - x_{\bar{i}}) - b_k(y_j - y_{\bar{j}}) + 1 < M^2_{ijk} l^k_{ij}, \quad \forall i,j,k; \\
&\quad \sum_{k=1}^{n} l^k_{ij} \ge n z_{ij}, \quad \forall i,j; \\
&\quad \sum_{k=1}^{n} l^k_{ij} \le (n-1) + z_{ij}, \quad \forall i,j; \\
&\quad z_{\bar{i}\bar{j}} = 1; \\
&\quad \sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij} z_{ij} \ge \alpha; \\
&\quad a_k, b_k \in \mathbb{R}, \quad \forall k; \\
&\quad l^k_{ij} \in \{0,1\}, \quad \forall i,j,k; \\
&\quad z_{ij} \in \{0,1\}, \quad \forall i,j.
\end{aligned}
\tag{19}
$$

V. 求解方法

本节将提出一种启发式算法，以高效求解前述混合整数非线性规划优化问题。

A. MINLP最优算法

尽管优化框架式（19）是一个具有非线性约束的MINLP 问题，但除式（9）和式（10）外，所有约束实际上都是线性约束。幸运的是，即使二次约束式（9）和式（10）也是双线性约束，且仅涉及互不重叠的变量对的乘积。Gurobi非常适合处理此类约束。它采用割平面和分支算法；更多信息请参见 [37]。借助该求解器，**算法2**求得式（19）的一个最优解，从而得到由算法1计算的概率可达集的凸近似。

B. MINLP启发式算法

然而，算法2计算开销较大，且在网格点数量较多时扩展性较差。为解决这一问题，我们开发了**算法3** ，可在确保准确性的同时高效求解式（19）。算法2 仅作为基准，用于提供式（19）的最优解

算法2 MINLP最优算法

1: **函数** MINLPSOLVER(α, $\boldsymbol{g}$, $\boldsymbol{w}$)
2: $\bar{i}$, $\bar{j}$ = 最大化($\boldsymbol{w}$)
3: 根据 $x_{\bar{i}}$, $y_{\bar{j}}$ 确定 $\bar{i}$, $\bar{j}$ 和 $\boldsymbol{g}$
4: 为网格点制定方程（19），权重为 $\boldsymbol{g}$ 为 $\boldsymbol{w}$ 给定置信水平 α
5: 实施割平面和分支算法 求解方程（19）以求得 a_k, b_k
6: **返回** a_k, b_k, $x_{\bar{i}}$, $y_{\bar{j}}$

7: **函数** FINDPOLYGON($\boldsymbol{g}$, $\boldsymbol{w}$, α)
8: a_k, b_k, $x_{\bar{i}}$, $y_{\bar{j}}$ = MINLPSOLVER(α, $\boldsymbol{g}$, $\boldsymbol{w}$)
9: **返回** a_k, b_k, $x_{\bar{i}}$, $y_{\bar{j}}$

10: FINDPOLYGON($\boldsymbol{g}$, $\boldsymbol{w}$, α)

，仅用于比较。如图11所示，该算法根据核密度估计值进行加权抽样，从所有网格点中选取代表性网格点 $\boldsymbol{g}'$。 $\boldsymbol{g}$与使用所有网格点 $\boldsymbol{g}$的原始 MINLP 问题不同，我们可以使用代表性网格点 $\boldsymbol{g}'$构建一个新的 MINLP 问题。然后，可以使用 Gurobi 中的割平面和分支算法求解这个新构建的 MINLP 问题。如图12所示，新 MINLP 问题的最优解是原始 MINLP 问题解的近似。通过这种方式，效率的提升是以牺牲最优性为代价的。由于 $\boldsymbol{g}'$ 的规模可能远小于 $\boldsymbol{g}$，新 MINLP 问题减少了决策变量和约束的数量，从而大幅缩短计算时间。

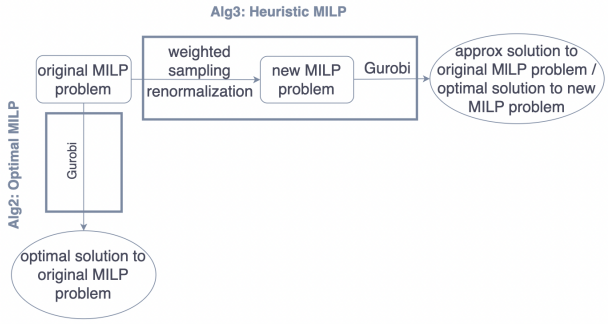


图11. 混合整数非线性规划启发式算法与混合整数非线性规划最优解之间的流程差异

在算法 3的伪代码中，我们使用 N_s 表示代表性网格点的数量 $\boldsymbol{g}'$。从第 1 行到第 6 行，我们使用 [38]提出的AES算法随机选择 N_s 个代表性网格点 $\boldsymbol{g}'$ ，从全部 N^2 网格点 $\boldsymbol{g}$ 中按照其归一化权重进行无放回抽样 $\boldsymbol{w}$。回顾一下， $\boldsymbol{g}$ 是算法1 的输出，而 $\boldsymbol{w}$ ：$= [w_{ij}]$ 通过公式（6）获得。从第 7 行到第 9 行，代表性网格点的原始权重 $\boldsymbol{w}'$ $\boldsymbol{g}'$ 被重新归一化为$\hat{}\boldsymbol{w}'$，使其总和等于 1。当3算法获得

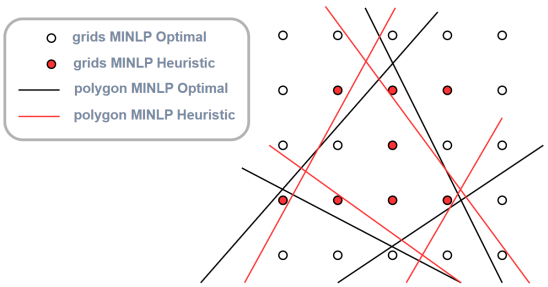


图12. 混合整数非线性规划启发式算法与 MINLP 最优之间的结果差异

与实施算法 2后的解相比。

算法3 MINLP启发式算法

函数 WGTSAMP($\boldsymbol{g}$, $\boldsymbol{w}$, N_s)
  $u_{ij} =$ 随机(0, 1)
  $k_{ij} = u_{ij}^{1/w_{ij}}$
  按 k_{ij}的降序取 $\boldsymbol{g}$ 中的前 N_s 个网格
  作为代表性网格 $\boldsymbol{g}'$
  取那些 N_s 网格的权重，来自 $\boldsymbol{w}$
  作为代表性权重 $\boldsymbol{w}'$
  返回 $\boldsymbol{g}'$, $\boldsymbol{w}'$

函数 NORMWGT($\boldsymbol{w}'$, N_s)
  $\hat{\boldsymbol{w}}' = \boldsymbol{w}' + \text{ones}(N_s) \cdot (1 - \text{sum}(\boldsymbol{w}'))/N_s$
  返回 $\hat{\boldsymbol{w}}'$

函数 MINLPSOLVER($\boldsymbol{g}'$, $\hat{\boldsymbol{w}}'$, α, $\boldsymbol{g}$)
  $\bar{i}, \bar{j} = \arg\max(\hat{\boldsymbol{w}}')$
  根据 i, j 和 $\boldsymbol{g}$确定 $x_{\bar{i}}$, $y_{\bar{j}}$
  针对网格点构建公式（19） $\boldsymbol{g}'$ 权重为 $\hat{\boldsymbol{w}}$
  给定置信水平 α
  实现割平面和分支算法
  求解方程（19）以获得 a_k, b_k
  返回 a_k, b_k, $x_{\bar{i}}$, $y_{\bar{j}}$

16：函数 FINDPOLYGON($\boldsymbol{g}$, $\boldsymbol{w}$, N_s, α) 17： $\boldsymbol{g}'$, $\boldsymbol{w}'$ = **WGTSAMP($\boldsymbol{g}$, $\boldsymbol{w}$, N_s) 18：** $\hat{\boldsymbol{w}}'$ = **NORMWGT($\boldsymbol{w}'$, N_s) 19：** a_k, b_k, $x_{\bar{i}}$, $y_{\bar{j}}$ = **MINLPSOLVER($\boldsymbol{g}'$, $\hat{\boldsymbol{w}}'$, α, $\boldsymbol{g}$) 20：根据** a_k, b_k, $x_{\bar{i}}$, $y_{\bar{j}}$**确定多边形** S **21：返回** S

22：FINDPOLYGON($\boldsymbol{g}$, $\boldsymbol{w}$, N_s, α)

VI. 主要结果

在本节中，我们通过全面的案例研究，比较混合整数非线性规划最优算法（算法2）和混合整数非线性规划启发式算法（算法3）与 [28], 中介绍的包围盒算法的性能，并研究不同参数对这些算法性能的影响。我们还讨论了混合整数非线性规划启发式算法的鲁棒性。测试在 Python 3.9 上进行，使用的设备为配备 64GB内存的台式机，处理器为 Intel(R) Core(TM)i9-12900KF，3187兆赫，16 个核心、24 个逻辑处理器。

案例研究分为两个阶段：

1) 求解阶段：根据核密度估计值以及通过运行算法 M 1从一组数据样本中得到的概率可达集，我们构建一个 MINLP 问题。随后，分别采用混合整数非线性规划最优解和 MINLP 启发式算法来求解概率可达集的凸近似。作为对比，采用包围盒算法求出一个包围概率可达集的凸四边形；

2) 测试阶段：给定上一阶段得到的凸多边形，我们可以生成一组新的 $N_{\text{test}}(N_{\text{test}} \gg M)$ 个数据样本，并计算落在多边形内的数据样本数量与生成的数据样本总数之比。随着 N_{test} 增大，该比率将收敛于系统状态位于多边形内的真实概率。

A. 案例设置

1) 情况 *I*：在此案例中，我们考虑一个在平面上运动的 Dubins车辆模型。车辆的速度服从截断高斯分布 $v \sim \mathcal{N}(\mu_1, \sigma_1)$, 均值为 $\mu_1 = 190\,\text{km/h}$，标准差为 $\sigma_1 = 5\text{km/h}$，且速度取值位于区间$[165, 220]\,\text{km/h}$ 内。车辆的航向角服从另一截断高斯分布 $\theta \sim \mathcal{N}(\mu_2, \sigma_2)$，均值为$\mu_2 = 10\text{degs}$，标准差为 $\sigma_2 = 30$ degs，且航向角取值位于区间 $[-50, 70]$ degs 内。

车辆在时间 k 的位置 $\boldsymbol{x}_k := [x_k, y_k]^\top$ 可由离散时间动态系统建模

$$\begin{bmatrix} x_k \\ y_k \end{bmatrix} = \begin{bmatrix} x_{k-1} \\ y_{k-1} \end{bmatrix} + \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} v\Delta t,$$

其中，$\boldsymbol{x}_{k-1} := [x_{k-1}, y_{k-1}]^\top$ 表示车辆在时间 $k-1$ 的位置，速度 v 和航向角 θ 假设在时间间隔 $\Delta t = 1$ 秒内保持恒定。由于车辆的 v 和 θ 存在不确定性，位置$\boldsymbol{x}_k$ 是一个服从非高斯分布的随机向量，我们称之为扇形分布。图13 中的阴影区域表示车辆在某一时刻的位置可达集。

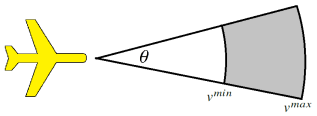


图1 车辆在某一时刻的位置可达集

2）情况 *II*：在这种情况下，我们展示车辆在某一时刻于平面上的可能位置（数据样本）的散点图 $[x, y]^\top$，该图由 x 和 y 各自的边缘直方图生成。如图 14所示，车辆在该时刻的位置服从双峰分布，因此是非高斯分布。

在下文中，公式（**2**）中的带宽矩阵 H 设为$\left[\begin{smallmatrix} 0.2 & 0 \\ 0 & 0.2 \end{smallmatrix}\right]$, 参数 10^4 采用一个通用常数

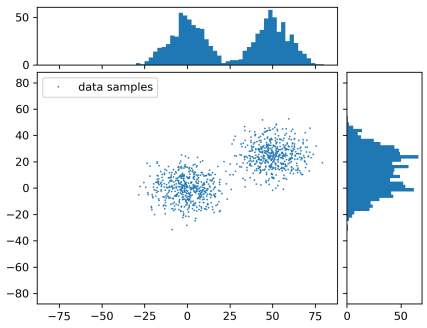


图14. 由两个边缘直方图生成的联合双峰分布

M_{ijk}^1，M_{ijk}^2公式（13）和公式（14）中的参数，以及数据样本数量设为 $M = 1000$。以下设置选作基线：边数为 $n = 4$，置信水平为$\alpha = 90\%$，混合整数非线性规划最优解使用的网格点数量为 $N^2 = 20^2$，混合整数非线性规划启发式算法使用的网格点数量在情况 I 中为 $N_s = 70$，在情况 II 中为 $N_s = 60$。在以下图表中，基线用符号“*”标记。

B. 不同边数

在本部分中，我们比较不同多边形边数下不同算法的性能：$n = 3$，4，5。其他参数与基线相同。

结果如图 15和表I所示。请注意，边数 n的改变仅适用于混合整数非线性规划最优解和MINLP启发式算法。对于混合整数非线性规划最优解或MINLP启发式算法，随着 n 从 3 增加到 5，多边形面积减小，而比率仅略有变化，且始终更接近置信水平 α。这表明，在确保准确性的同时，两种算法得到的结果都变得不那么保守。然而，随着式（19）中的决策变量和约束数量随 n的增加而增加，相应的计算时间也会增加。为平衡最优性和计算效率，我们将 $n = 4$。

当 $n = 4$时，混合整数非线性规划最优解或混合整数非线性规划启发式算法得到的面积和比率都远小于包围盒算法的结果，这表明这两种算法得到的多边形比包围盒算法得到的多边形更紧凑、准确。这显然有利于实际运动规划。在考虑车辆与障碍物之间的碰撞规避时，较不保守的概率可达集凸近似可提供更大的可行搜索区域，从而有望规划出更优的轨迹。此外，MINLP启发式算法的计算时间远短于混合整数非线性规划最优解（情况 I 为 0。801s < 70.5 s，情况 II 为 0。361 s < 18.6s），但两者在面积和比率上的结果仍然相近。这证明了MINLP启发式算法在确保准确性的同时具有较高的计算效率，使在线实现成为可能。

如第V节所述，通过混合整数非线性规划最优解得到的多边形，在不违反约束的意义下是最优的，因为其面积最小。然而，对于某些情况，表I中，混合整数非线性规划启发式算法的面积（例如 1807。9 米^2）可能略小于混合整数非线性规划最优解（例如 1827。0米^2）。近似误差来自使用新选取的 N_s 网格点 $\boldsymbol{g}'$进行重新归一化。通过混合整数非线性规划启发式算法得到的多边形，是基于所选 N_s 网格点 N^2的新 MINLP 问题的最优解，该问题也是对涉及 N^2网格点 N^2的原始 MINLP 问题的近似。由于原始 MINLP 问题与新 MINLP 问题采用了不同的网格设置，因此混合整数非线性规划启发式算法得到的多边形面积可能小于混合整数非线性规划最优解。然而，根据我们的实验，这些误差相对较小。

C. 不同数量的网格点

在此，我们比较采用不同数量网格点时各算法的性能：$N^2 = 10^2$，20_2，30_2，40_2。对于混合整数非线性规划启发式算法，N_s =20，70，160，280 个网格点分别针对情况 I 随机选取；对于情况 II，则选取 $N_s = 15$，60，135，240 个网格点。其他参数与基线相同。

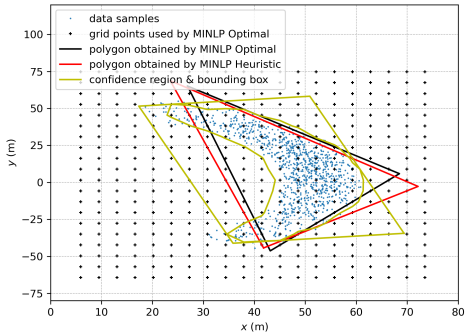
根据图 16 和表 II，我们可以得出如下结论。图 16a 和图 16b 中由三种算法实现得到的结果，与图 16 中其他子图的结果在视觉上存在很大差异。这是因为当网格点数量过少（$N^2 = 10^2$）时，核密度估计无法很好地近似概率密度函数。如表 II 所示，对于每种算法，随着 N^2 从 20^2 变化到 40^2,，比率和面积方面的性能均略有提升。然而，随着 N^2 增加，决策变量和约束数量的增加会导致计算时间显著增加。如表 II 所示，混合整数非线性规划启发式算法在情况 I 中的计算时间从 0.801 s 增加到 125.5 s，在情况 II 中则从 0.361s 增加到 23.8s。因此，准确性和最优性略有提升，但效率的代价极高。为实现折中，网格点数量可选择为$N^2 = 20^2$。

当 N^2 固定为 20^2, 时，混合整数非线性规划最优解和混合整数非线性规划启发式算法的面积和比率均远小于包围盒算法，这表明混合整数非线性规划最优解和混合整数非线性规划启发式算法在最优性和准确性方面显著优于包围盒算法。此外，对于混合整数非线性规划最优解和混合整数非线性规划启发式算法，其比率和面积结果相当接近。然而，混合整数非线性规划启发式算法的计算时间远优于混合整数非线性规划最优解，即在情况 I 中为 0.801s < 70.5s，在情况 II 中为 0.361s < 18.6s，这证明了混合整数非线性规划启发式算法的效率。因此，混合整数非线性规划启发式算法能够同时保证近最优性、准确性和效率，使其非常适合实时实现。

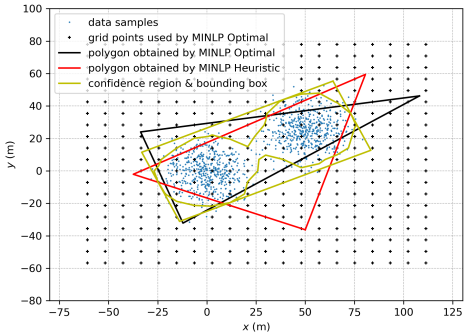
D. 不同的置信水平

在本部分中，我们比较算法在不同置信水平下的性能：α = 90%、95%和99%。其他参数与基线相同。

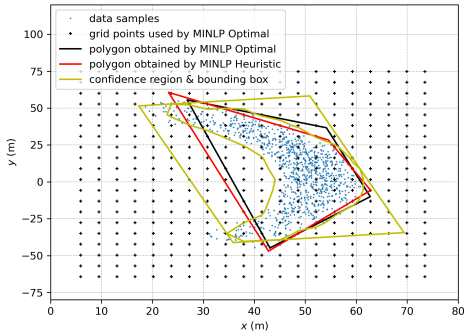
在图 17 和表 III中，对于情况 I 和 II，随着 α 的增加，分别通过混合整数非线性规划最优解、混合整数非线性规划启发式算法或包围盒得到的多边形面积也随之增大，而各算法的计算时间则略有波动。



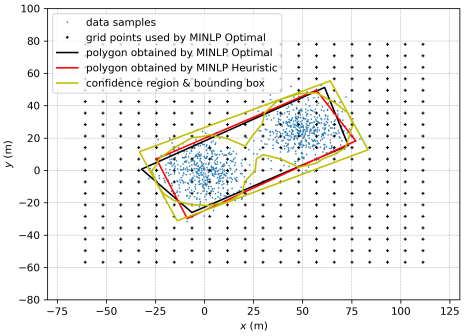
(a) 情况 I： $n = 3$



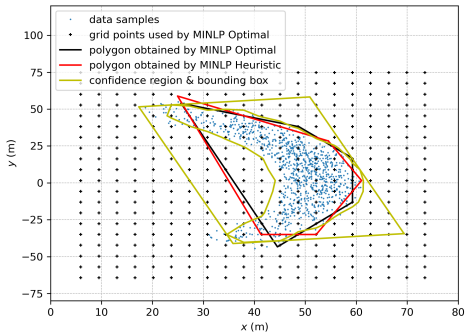
(b) 情况 II： $n = 3$



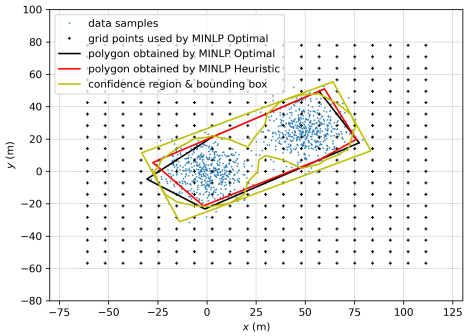
(c) 情况 I： $n = 4$ *



(d) 情况 II： $n = 4$ *



(e) 情况 I： $n = 5$

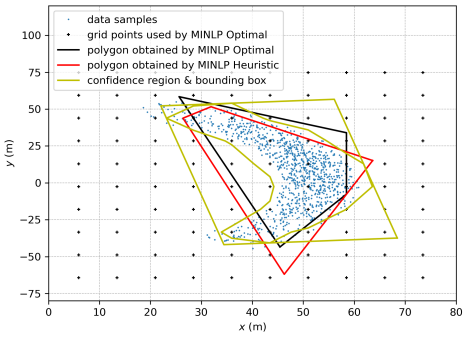


(f) 情况 II： $n = 5$

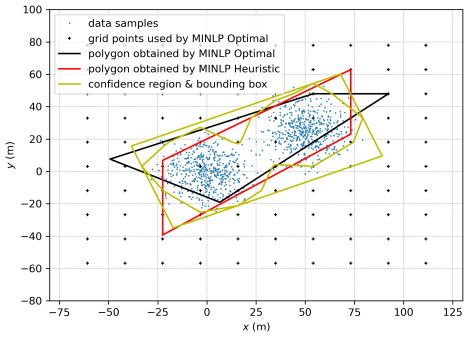
图15. 不同边数的比较 n （用于包围盒 $n \equiv 4$）

表I 不同边数的比较 n （用于包围盒 $n \equiv 4$）

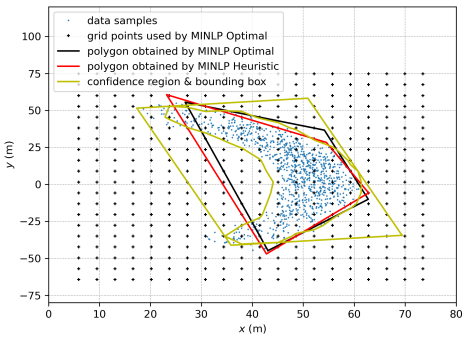
#边数 n	**算法**	**情况 I**			**情况 II**		
		比率	**面积（米^2）**	**时间（秒）**	**比率**	**面积（米^2）**	**时间（秒）**
3（三角形）	混合整数非线性规划最优解	90.1%	1835.9	27.5	89.1%	4220.7	13.7
	混合整数非线性规划启发式算法	90.2%	1911.9	0.533	90.1%	4706.6	0.209
	包围盒	N/A	N/A	N/A	N/A	N/A	N/A
4（四边形）*	混合整数非线性规划最优解	91.5%	1827.0	70.5	91.4%	3570.5	18.6
	混合整数非线性规划启发式算法	90.7%	1807.9	0.801	90.6%	3457.1	0.361
	包围盒	97.9%	3235.1	0.216	97.7%	4987.2	0.189
5（五边形）	混合整数非线性规划最优解	89.7%	1667.9	316.3	91.3%	3511.6	27.1
	混合整数非线性规划启发式算法	90.2%	1721.9	1.145	90.5%	3401.7	0.525
	包围盒	N/A	N/A	N/A	N/A	N/A	N/A



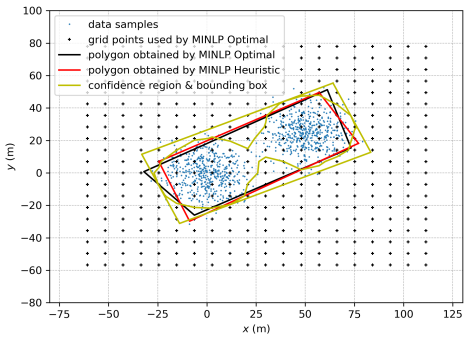
(a) 情况 I: $N^2 = 10^2$



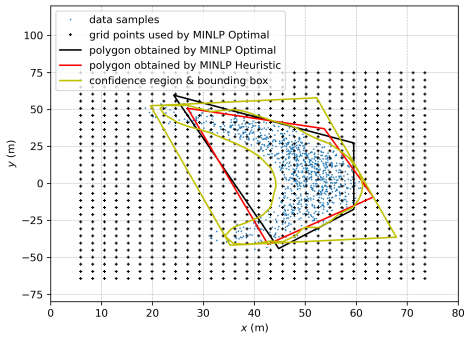
(b) 情况 II: $N^2 = 10^2$



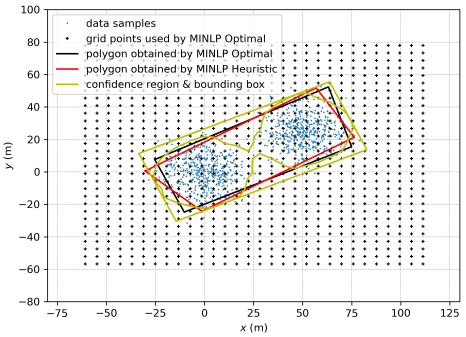
(c) 情况 I: $N^2 = 20^2$ *



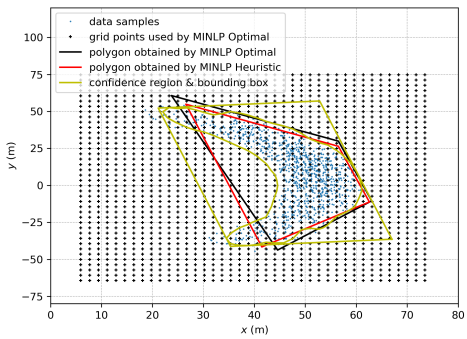
(d) 情况 II: $N^2 = 20^2$ *



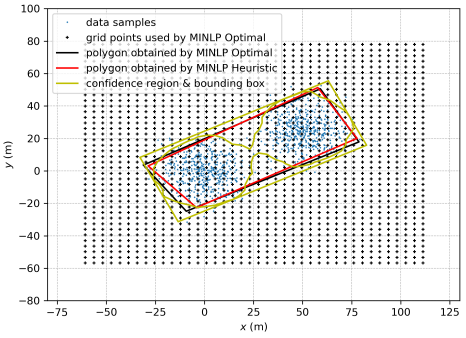
(e) 情况 I: # $N^2 = 30^2$



(f) 情况 II: $N^2 = 30^2$



(g) 情况 I: $N^2 = 40^2$

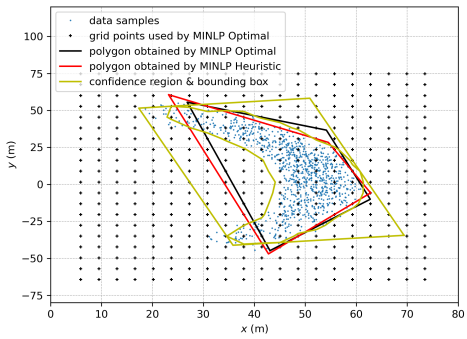


(h) 情况 II: # $N^2 = 40^2$

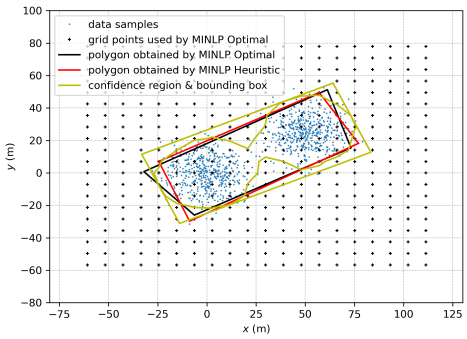
图 16. 不同网格点数量的比较 N^2

表 II 不同网格点数量的比较

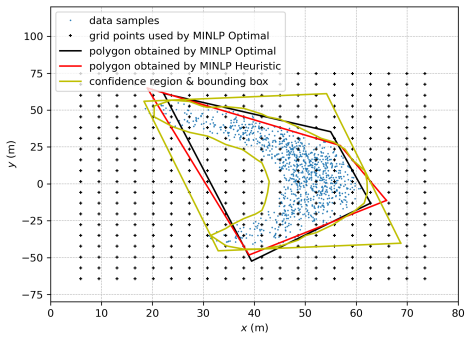
#网格点 N^2	算法	情况 I			情况 II		
		比率	面积（m^2）	时间（s）	比率	面积（m^2）	时间（s）
10^2	混合整数非线性规划最优解	87.2%	1703.6	0.321	82.2%	3784.7	0.082
	混合整数非线性规划启发式算法	90.2%	1908.7	0.064	92.6%	4110.5	0.015
	包围盒	97.9%	3257.2	0.209	99.6%	6318.8	0.184
20^2 *	混合整数非线性规划最优解	91.5%	1872.0	70.5	91.4%	3570.5	18.6
	混合整数非线性规划启发式算法	90.7%	1807.9	0.801	90.6%	3457.1	0.361
	包围盒	97.9%	3235.1	0.216	97.7%	4987.2	0.189
30^2	混合整数非线性规划最优解	91.4%	1829.1	1150.2	92.0%	3578.3	362.9
	混合整数非线性规划启发式算法	90.4%	1755.9	8.9	91.8%	3443.1	6.9
	包围盒	97.7%	3157.7	0.225	97.6%	4893.5	0.193
40^2	混合整数非线性规划最优解	91.3%	1827.0	2360.1	91.4%	3554.8	1237.4
	混合整数非线性规划启发式算法	90.1%	1721.1	125.5	90.9%	3425.0	23.8
	包围盒	97.1%	3030.2	0.239	97.3%	4722.6	0.211



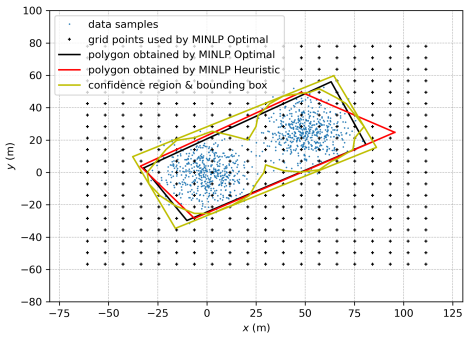
(a) 情况 I： $\alpha = 90\%$ *



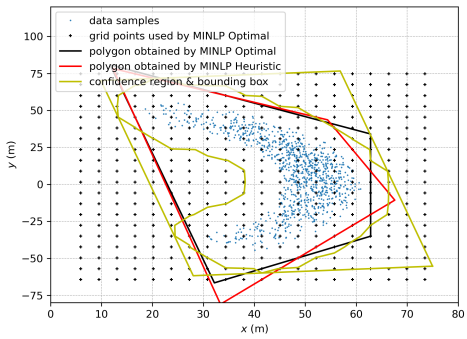
(b) 情况 II： $\alpha = 90\%$ *



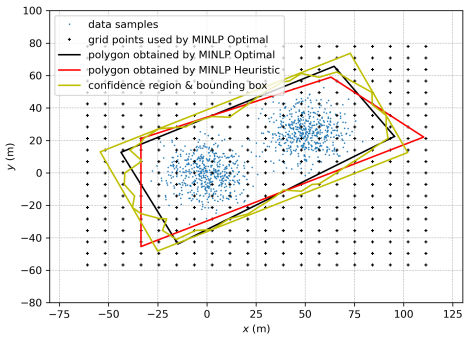
(c) 情况 I： $\alpha = 95\%$



(d) 情况 II： $\alpha = 95\%$



(e) 情况 I： $\alpha = 99\%$



(f) 情况 II： $\alpha = 99\%$

图17. 不同置信水平的比较 α

表 III 不同置信水平的比较

置信水平 α	算法	情况 I			情况 II		
		比率	面积（m^2）	时间（s）	比率	面积（m^2）	时间（s）
90% *	混合整数非线性规划最优解	91.5%	1872.0	70.5	91.4%	3570.5	18.6
	混合整数非线性规划启发式算法	90.7%	1807.9	0.801	90.6%	3457.1	0.361
	包围盒	97.9%	3235.1	0.216	97.7%	4987.2	0.189
95%	混合整数非线性规划最优解	95.7%	2359.6	53.5	96.1%	4312.6	8.8
	混合整数非线性规划启发式算法	96.1%	2424.7	0.417	96.2%	4471.5	0.112
	包围盒	99.4%	3709.1	0.243	99.0%	5613.5	0.235
99%	混合整数非线性规划最优解	99.9%	4350.6	50.6	99.9%	7223.7	7.8
	混合整数非线性规划启发式算法	99.9%	4443.5	0.377	99.9%	7460.1	0.090
	包围盒	100%	6314.8	0.303	100%	9539.5	0.271

当 α 固定时，通过混合整数非线性规划最优解得到的多边形面积远小于包围盒的面积。这意味着，与包围盒相比，通过混合整数非线性规划最优解得到的多边形保守性要低得多。然而，实施混合整数非线性规划最优解所需的计算时间远长于包围盒。也就是说，混合整数非线性规划最优解不适合进行在线评估。相比之下，通过混合整数非线性规划启发式算法得到的多边形面积接近混合整数非线性规划最优解，而实施混合整数非线性规划启发式算法所需的计算时间足够短。因此，混合整数非线性规划启发式算法在保证准确性的同时，运行速度远快于混合整数非线性规划最优解。

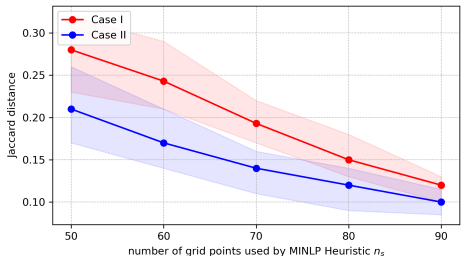
E. 混合整数非线性规划启发式算法的鲁棒性

如算法 3所示，每次运行 MINLP 启发式算法时，我们都会对网格点进行加权抽样，以选出具有代表性的网格点 $\boldsymbol{g}'$，这些网格点取自 $\boldsymbol{g}$。由于加权抽样具有随机性，每次选出的网格点 $\boldsymbol{g}'$ 通常都不相同，因此每次应用 MINLP 启发式算法得到的多边形也会发生变化。这引发了以下关于 MINLP 启发式算法鲁棒性的问题。1）空间：指每次得到的凸多边形是否会有显著差异；2）时间：指每次所花费的计算时间是否会有显著差异。若存在重大差异，则意味着 MINLP 启发式算法对于真实场景中的应用而言不够鲁棒。

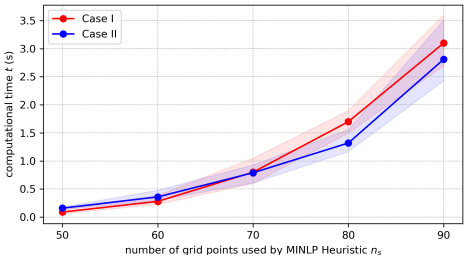
两个多边形之间的相似性可以用杰卡德距离来量化。形式上，对于任意两个集合 A 和 B，杰卡德距离定义为 $d(A,\ B) = 1 - \frac{\text{基数}(A \cap B)}{\text{基数}(A \cup B)}$ [33]，其中，运算 $\overline{\text{Card}}$ 用于求集合的基数。杰卡德距离越小，两个集合就越相似。

我们分析了混合整数非线性规划启发式算法在上述案例 I 和 II 中，针对其使用的网格点数量 N_s $\boldsymbol{g}'$ 变化时的鲁棒性。对于情况 I 或情况 II，除数量 N_s外，所有参数均保持不变，且与相应的基线相同。

对于情况 I 或情况 II，给定固定的数量 N_s，我们运行一次混合整数非线性规划最优解，以获得一个固定的最优多边形；并运行混合整数非线性规划启发式算法十次，以获得十个近似多边形，这些多边形每次可能都不同。我们计算每个近似多边形与最优多边形之间的杰卡德距离，最终得到十个距离。此外，我们还计算每次运行



(a) 空间鲁棒性



(b) 时间鲁棒性

图 18. 混合整数非线性规划启发式算法的鲁棒性分析 hm

混合整数非线性规划启发式算法所需的计算时间。我们分别针对 $N_s = 50$、60、70、80、90 重复上述操作。结果如图 18 所示。如图 18a 所示，对于两种情况，当 N_s 太小时（例如 $N_s = 50$），不同的杰卡德距离会在均值附近大幅波动，这表明在这种情况下，混合整数非线性规划启发式算法的鲁棒性较差。随着 N_s 从 50 增至 90，杰卡德距离的均值分别从 0.28、0.21 降至 0.12、0.10。这意味着，随着 N_s 增大，混合整数非线性规划启发式算法得到的近似多边形会越来越接近最优解。此外，十个杰卡德距离的方差也会随着 N_s 增大而减小。也就是说，随着 N_s 增大，混合整数非线性规划启发式算法的鲁棒性会增强。如图 18b 所示，随着 N_s 增大，情况 I 的平均计算时间从 0.09 s 增至 3.1 s，情况 II 的平均计算时间从 0.16 s 增至 2.81 s。此外，随着 N_s 增大，计算时间的方差也会增大。

总之，MINLP启发式算法的鲁棒性是以牺牲计算效率为代价的。对于所考虑的案例研究，$N_s = 70$ 对于情况 I 或 $N_s = 60$ 对于情况 II 实现了平衡。这使得 MINLP 启发式算法

足够鲁棒，且其应用前景广阔。

VII. 结论

本文提出了一种对不确定动态系统的概率可达集进行高效凸近似的方法。针对任意未知的不确定性，尤其是无界不确定性，本文引入了概率可达集这一概念，将机会约束表述与可达性分析联系起来，作为对传统可达集的扩展。在本文中，概率可达集通过采用核密度估计方法的数据驱动方法获得。随后对快速傅里叶变换进行定制，以加速核密度估计的计算，使其具有较高的计算吸引力。概率可达集的不规则性或非凸性限制了其在最优设计应用中的使用。为解决这一问题，我们构建了一个 MINLP 问题，其解能够获得概率可达集的凸近似。随后，我们开发了一个MINLP启发式算法来求解该问题。全面的案例研究表明，所提出的算法在为概率可达集提供凸近似的同时，兼具准确性、效率、近最优性以及鲁棒性等优势。

本文仅研究了二维系统。未来的工作将致力于解决更高维问题，并将研究结果应用于不确定机器人系统安全关键实时运动规划算法的设计。

VIII. 附录

A. 引理1的证明

I) 首先，我们证明，任意两种形式上不同的直线表示对应不同的直线。

给定 $n \geq 3$ 条不经过原点的平面直线l^k: $= a_k x + b_k y - 1 = 0$, $k \in \{1, \ldots, n\}$，我们可以构造一个直线序列（l^1, l^2, ..., l^n, l^1）。当且仅当 $j = i^{\oplus} \vee j = i^{\ominus}$ 时，序列中形式上不同的两条直线 l^i, l^j 被定义为相邻。由于有 n 个交点 $V_{ij} := l^i \cap l^j$, $i \in \{1, \ldots, n\}$, $j = i^{\oplus}$，因此 $l^i \neq l^j$, $i \in \{1, \ldots, n\}$, $j = i^{\oplus}$。由此，序列中任意两条形式上不同且相邻的直线确实彼此不同。

对于 $n = 3$，直线序列为（l^1, l^2, l^3, l^1）。由于序列中的任意两条形式上不同的直线（l^1, l^2, l^3, l^1）都相邻，因此 l^1, l^2, l^3 确实彼此不同。对于 $n \geq 4$，序列中至少存在一对形式上不同且不相邻的直线。我们已经证明，序列中任意两条形式上不同且相邻的直线确实彼此不同，因此还需证明，在本引理的条件下，序列中任意两条形式上不同且不相邻的直线也确实彼此不同。我们用反证法证明这一点。假设 l^u, l^v, $u \in \{1, \ldots, n\}$, $v \in \{1, \ldots, n\} - \{u^{\ominus}, u, u^{\oplus}\}$ 是两条形式上不同且不相邻的直线，但实际上是同一条直线。那么 $V_{v^{\ominus}v}$ 和 $V_{vv^{\oplus}}$ 都在直线 l^u 上，这意味着

$$\begin{aligned} &-a_u\left(b_{v^{\ominus}} - b_v\right) + b_u\left(a_{v^{\ominus}} - a_v\right) - \left(a_{v^{\ominus}} b_v - b_{v^{\ominus}} a_v\right) = 0, \\ &-a_u\left(b_v - b_{v^{\oplus}}\right) + b_u\left(a_v - a_{v^{\oplus}}\right) - \left(a_v b_{v^{\oplus}} - b_v a_{v^{\oplus}}\right) = 0. \end{aligned} \tag{20}$$

然而，由于 $v \notin \{u^{\ominus}, u, u^{\oplus}\}$, $V_{v^{\ominus}v}$ 的下标在形式上与 $V_{u^{\ominus}u}$ 和 $V_{uu^{\oplus}}$ 不同，并且下标 $V_{vv^{\oplus}}$ 在形式上也不同于 $V_{u^{\ominus}u}$ 和$V_{uu^{\oplus}}$。因此，由于约束式（10）得到满足，我们获得了两个实例

$$\begin{aligned} &-a_u\left(b_{v^{\ominus}} - b_v\right) + b_u\left(a_{v^{\ominus}} - a_v\right) - \left(a_{v^{\ominus}} b_v - b_{v^{\ominus}} a_v\right) < 0, \\ &-a_u\left(b_v - b_{v^{\oplus}}\right) + b_u\left(a_v - a_{v^{\oplus}}\right) - \left(a_v b_{v^{\oplus}} - b_v a_{v^{\oplus}}\right) < 0, \end{aligned}$$

这与公式（20）矛盾。

因此，如果满足约束公式（9）和公式（10），那么任意两条形式上不同的直线实际上彼此不同。

II) 接下来，我们证明，如果存在 n 交点$V_{ij} := l^i \cap l^j$、$i \in \{1, \ldots, n\}$、$j = i^{\oplus}$（即 $D_{ij} := a_i b_j - b_i a_j > 0$、$i \in \{1, \ldots, n\}$、$j = i^{\oplus}$根据公式（9））, 并且满足约束公式（10），那么任意两个形式上不同的交点实际上彼此不同。

我们可以通过反证法证明这一点。假设存在两个形式上不同的交点 V_{ij}、V_{uv}、$i \in \{1, \ldots, n\}$、$j = i^{\oplus}$、$u \in \{1, \ldots, n\} - \{i\}$、$v = u^{\oplus}$，它们实际上是同一个交点。由于 $n \geq 3$，因此可得 $(u \neq i \wedge u \neq j) \vee (v \neq i \wedge v \neq j)$，不失一般性地，假设 $v \neq i \wedge v \neq j$。因此，约束公式（10）适用于 V_{ij} 和 l^v，从而得到一个实例

$$-a_v\left(b_i - b_j\right) + b_v\left(a_i - a_j\right) - \left(a_i b_j - b_i a_j\right) < 0. \tag{21}$$

然而，由于 V_{ij}, V_{uv} 确实是同一个交点， V_{ij} 位于直线 l^v 上，这意味着

$$-a_v\left(b_i - b_j\right) + b_v\left(a_i - a_j\right) - \left(a_i b_j - b_i a_j\right) = 0,$$

这与式（21）矛盾。

结合 I) 和 II)，我们得出：若满足约束式（9）和式（10），则任意两个形式上不同的交点（分别地，直线）表示不同的交点（分别地，直线），因此所有 n交点（分别地，直线）彼此确实不同。

B. 定理1的证明

I) 由于共有 $n \geq 3$ 个交点 $V_{ij} := l^i \cap l^j$, $i \in \{1, \ldots, n\}$, $j = i^{\oplus}$，因此我们可以将它们枚举为交点序列 $S_g := (V_{12}, V_{23}, \ldots\ldots, V_{(n-1)n}, V_{n1}, V_{12})$。根据引理 1，由于这些直线的系数满足约束式（9）和式（10），因此任意两个形式上不同的交点（分别地，直线）彼此确实不同。

II）我们还可以构造一个有限符号序列$S_s :=$（“V_{12}”、“V_{23}”.....、“$V_{(n-1)n}$”、“V_{n1}”、“V_{12}”）。在 S_s中，1）由于 $n \geq 3$，至少存在三个形式不同的符号“V_{12}”、“V_{23}”和“V_{31}”；2）除仅有第一个符号“V_{12}”与最后一个符号“V_{12}”形式相同的情况外，没有任何符号出现超过一次。因此，定义 9 适用，且 S_s 是形式包围的。

考虑一个从 f_{tag} 到 S_s 的映射 S_g，该映射通过标记交点的操作给出，即 $V_{ij} := l^i \cap l^j$, $i \in \{1, \ldots, n\}$, $j = i^{\oplus}$。可以验证， f_{tag}是单射、满射且保序的。映射 f_{tag} 还满足定义 10中的条件 4)。这是因为在 S_s中，对于任意

两个连续符号“$V_{i\ominus i}$” 和 “$V_{ii\oplus}$”，以及任意第三个符号 “V_{uv}”，其形式上不同于 “$V_{i\ominus i}$” 和 “$V_{ii\oplus}$”，根据公式 (10) 可得

$$-a_i\left(b_u-b_v\right)+b_i\left(a_u-a_v\right)-\left(a_ub_v-b_ua_v\right)\neq 0,\\ u\notin\{i^{\ominus},i\},v=u^{\oplus},$$

这意味着交点 V_{uv} 不在直线l^i上。根据 I)，由于形式不同的符号 “$V_{i\ominus i}$”和 “$V_{ii\oplus}$” 指向不同的交点 $V_{i\ominus i}$ 和 $V_{ii\oplus}$，因此这两个交点确定了一条非退化线段。此外，由这两点确定的线段是直线 l^i的一条线段。因此，第三个符号“V_{uv}” 所指向的交点不在边 $(V_{i\ominus i},\ V_{ii\oplus})$上。综上，映射 f_{tag} 满足定义 10中的所有条件，因此是 A型同构。

根据备注 1，由于 S_s 是形式包围的，且f_{tag} 是 A型同构，因此由 G形成的图 S_g 是包围的。

III) 由于 $n\geq 3$，S_s 中的形式不同符号的数量为 $n\geq 3$。

正如 II) 中得出的结论，映射 f_{tag} 是单射、满射且保序的。映射 f_{tag} 还满足定义 12中的条件 4)。这是因为，对于 S_s中的任意三个连续符号 “$V_{i\ominus i}$”、“$V_{ii\oplus}$”、“$V_{i^{\oplus}i^{\oplus\oplus}}$”，应用式 (10) 可得

$$-a_i\left(b_{i^{\oplus}}-b_{i^{\oplus\oplus}}\right)+b_i\left(a_{i^{\oplus}}-a_{i^{\oplus\oplus}}\right)-\left(a_{i^{\oplus}}b_{i^{\oplus\oplus}}-b_{i^{\oplus}}a_{i^{\oplus\oplus}}\right)\neq 0,$$

这意味着交点 $V_{i^{\oplus}i^{\oplus\oplus}}$ 不在直线 l^i上。由 I）可知，由于形式不同的符号“$V_{i\ominus i}$” 和“$V_{ii\oplus}$” 指代不同的交点 $V_{i\ominus i}$ 和 $V_{ii\oplus}$，因此 $V_{i\ominus i}$ 和 $V_{ii\oplus}$ 彼此不同。此外，交点 $V_{i\ominus i}$ 和 $V_{ii\oplus}$ 位于直线 l^i上。因此，三个交点 $V_{i\ominus i}$、$V_{ii\oplus}$、$V_{i^{\oplus}i^{\oplus\oplus}}$ 不共线。综上，映射 f_{tag} 满足定义 12 中的所有条件，因此是B型同构。

根据备注 2，由于 f_{tag} 是B型同构，且 $n\geq 3$ 中存在形式不同的符号 S_s，因此，由 S_g 形成的图是非退化的。

IV) 我们已经证明，由 G 形成的图 S_g 是一个包围的 n 边多边形。对于直线 l^k，$k\in\{1,\ldots,n\}$一旦满足约束式（10），即可推出

$$\begin{cases}-a_k\left(b_i-b_j\right)+b_k\left(a_i-a_j\right)-\left(a_ib_j-b_ia_j\right)<0,\\ i\in\{1,\ldots,n\}-\{k^{\ominus},k\},j=i^{\oplus},\\ \\ -a_k\left(b_i-b_j\right)+b_k\left(a_i-a_j\right)-\left(a_ib_j-b_ia_j\right)=0,\\ i\in\{k^{\ominus},k\},j=i^{\oplus}.\end{cases}\tag{22}$$

由于存在 n 个交点 V_{ij}，$i\in\{1,\ldots,n\}$，$j=i^{\oplus}$，根据式（9），可得 $D_{ij}:=a_ib_j-b_ia_j>0$，$i\in\{1,\ldots,n\}$，$j=i^{\oplus}$。因此，不等式式（22）可转换为

$$\begin{cases}a_k\frac{-(b_i-b_j)}{D_{ij}}+b_k\frac{(a_i-a_j)}{D_{ij}}-1<0,\\ i\in\{1,\ldots,n\}-\{k^{\ominus},k\},j=i^{\oplus},\\ \\ a_k\frac{-(b_i-b_j)}{D_{ij}}+b_k\frac{(a_i-a_j)}{D_{ij}}-1=0,\\ i\in\{k^{\ominus},k\},j=i^{\oplus},\end{cases}\tag{23}$$

这意味着所有 n 交点 V_{ij}，$i\in\{1,\ldots,n\}$，$j=i^{\oplus}$都位于直线 l^k的同一侧。因此，在 S_g中，对于任意两个相邻交点 V_{ij}，V_{uv}，其中 $i\in\{1,\ldots,n\}-\{k^{\ominus}\}$，$j=i^{\oplus}$，$u=i^{\oplus}$，$v=i^{\oplus\oplus}$，我们可以定义一个由 V_{ij}和 V_{uv} 的凸组合构成的集合，但不包括这两个边界点，该集合为

$$\Big\{t(-\frac{b_i-b_j}{D_{ij}},\frac{a_i-a_j}{D_{ij}})+(1-t)(-\frac{b_u-b_v}{D_{uv}},\frac{a_u-a_v}{D_{uv}}):\\ t\in\mathbb{R}\wedge 0<t<1\Big\}.$$

它是半平面 $\{(x,\ y):\ a_kx+b_ky-1<0\}$的子集。换句话说，边$(V_{ii\oplus},\ V_{i^{\oplus}i^{\oplus\oplus}})$ $i\in\{1,\ldots,n\}-\{k^{\ominus}\}$位于直线 l^k的严格内侧。这是因为

$$\begin{aligned}&a_k\left(t\frac{-(b_i-b_j)}{D_{ij}}+(1-t)\frac{-(b_u-b_v)}{D_{uv}}\right)+\\ &b_k\left(t\frac{(a_i-a_j)}{D_{ij}}+(1-t)\frac{(a_u-a_v)}{D_{uv}}\right)-1\\ =\ &t\left(a_k\frac{-(b_i-b_j)}{D_{ij}}+b_k\frac{(a_i-a_j)}{D_{ij}}\right)+\\ &(1-t)\left(a_k\frac{-(b_u-b_v)}{D_{uv}}+b_k\frac{(a_u-a_v)}{D_{uv}}\right)-1\\ <\ &0,\end{aligned}\tag{24}$$

根据式（23）。对于由 G构成的 S_g，根据式（24），我们已证明多边形的每条边$(V_{ii\oplus},\ V_{i^{\oplus}i^{\oplus\oplus}})$ $i\in\{1,\ldots,n\}-\{k^{\ominus}\}$都位于直线 l^k，$k\in\{1,\ldots,n\}$所定义的边$(V_{k^{\ominus}k},\ V_{kk^{\oplus}})$ $k\in\{1,\ldots,n\}$的同一侧。回顾凸多边形的定义：对于多边形的每条边，其余所有边都位于该边所定义直线的同一侧。因此，由 G 构成的 S_g 是一个凸多边形。

V) 回顾一下，对于任意点 (u,v) 以及由 n 条直线 $l^k:=a_kx+_nb_ky-1=0,k\in\{1,\ldots,n\}$ 确定的包围 n 边凸多边形，如果 $\bigwedge_{k=1}^{n}a_ku+b_kv-1<0$ 成立，则点 (u,v) 位于多边形内部。我们已在上文证明，若满足约束式（9）和式（10），则由 S_g 构成的图 G 是一个包围 n 边凸多边形。由于 $a_k\cdot 0+b_k\cdot 0-1\equiv -1<0,k\in\{1,\ldots,n\}$，原点（0,0）位于由 S_g构成的多边形 G 内部。

由此可得，若满足约束式 (9) 和式 (10)，则由 G 构成的图 S_g 是一个包围 n 边凸多边形，且原点(0,0)位于其内部。对于此类 $n\geq 3$ 平面直线 $l^k:=a_kx+b_ky-1=0$，$k\in\{1,\ldots,n\}$不经过原点且满足约束式 (9) 和式 (10) 的情形，由这些直线生成的区域 S（定义见式 (8)）的边界恰好是由 G 构成的图 S_g。由于该图 G 是一个包围 n

在满足式 (9) 和式 (10) 的条件下，边数为 S 的凸多边形包围原点，且原点位于其内部。因此，由这些直线生成的区域 n是一个包围原点且原点位于其内部的

推论*1*的证明

这可由使用（$\bar{x}$，$\bar{y}$）进行仿射坐标变换得到。更具体地说，给定坐标x-y，考虑变换（x^{new}，y^{new}）=（$x-\bar{x}$，$y-\bar{y}$）。在新坐标中，原点（0^{new}，0^{new}）位于（$\bar{x}$，$\bar{y}$）。此外，直线的表示变为

$$\left\{a(x-\bar{x})+b(y-\bar{y})-1=0 : a,b\in\mathbb{R}\wedge a^2+b^2\neq 0\right\}.$$

由定理1 可知，式（9）和式（10）足以保证区域 S:

$$\left\{ x^{\text{new}}\ y^{\text{new}} \quad \bigwedge_{k=1}^{n} a_k x^{\text{new}}+b_k y^{\text{new}}-1\leq 0 \right\} \qquad (\ ,\ ):$$

是一个包围n 边凸多边形，其原点 （0^{new}，0^{new}）位于其内部。变回原坐标系后，即可得到结论。

致谢

本研究得到美国国家科学基金会（NSF）项目 CMMI-2138612 的资助。本文所表达的任何意见、研究发现、结论或建议均为作者观点，不代表 NSF 的立场。作者谨向加州大学圣迭戈分校的 Lin Li、Paul Lathrop，以及圣迭戈州立大学的 Jun Xiang 致以谢意，感谢他们提供宝贵的意见并参与讨论。

参考文献

[1] T. Lew, A. Sharma, J. Harrison, A. Bylard 和 M. Pavone，“安全主动动力学学习与控制：一种序贯探索—利用框架”，《*IEEE*机器人学汇刊》，第38卷，第5期，第2888–2907页，2022年。[2] Z. Xing、H. Yue、T. Zhang、W. Wu 和 R. Hu，“基于粘合节点的异构、不同尺寸多移动机器人避碰与避让”，*IEEE*自动化科学与工程汇刊，2023年。[3] Y. Wen 和 P. Pagilla，“机器人操作臂的路径约束无碰撞最优轨迹规划”，*IEEE*自动化科学与工程汇刊，第20卷，第2期，第763–774页，2022年。[4] P. Wu、J. Xie、Y. Liu 和 J. Chen，“不确定条件下城市空中交通运行的风险有界公平性感知路径规划”，《航空航天科学与技术》，第127卷，文章编号107738，2022年。[5] A. Paudel、S. Gupta、M. Thapa、S. B. Mulani 和 R. W. Walters，“用于不确定性量化的高阶泰勒级数展开及高效局部灵敏度分析”，《航空航天科学与技术》，第126卷，107574，2022年。[6] P. Wu、X. Yang、P. Wei 和 J. Chen，“不确定环境下城市空中交通运行中具备空中间隔的安全在线引导”，*IEEE*智能交通系统汇刊，2022年。[7] F. Meng、L. Chen、H. Ma、J. Wang 和 M. Q.-H. Meng，“基于学习的环境不确定性下风险有界路径规划”，*IEEE*自动化科学与工程汇刊，2023年。[8] L. Zhang、X. Zhang、X. Chang 和 N. Zhao，“基于自适应容错控制的异构非线性奇异多智能体系统实时可达集综合（含不确定参数）”，*IEEE*自动化科学与工程汇刊，2023年。[9] Z. Li，“安全方法比较：控制屏障函数与可达性分析”，*arXiv*预印本 *arXiv:2106.13176*，2021年。[10] M. Althoff、G. Frehse 和 A. Girard，“可达性分析的集合传播技术”，《控制、机器人与自主系统年度评论》，第4卷，第369–395页，2021年。

[11] A. Devonport、F. Yang、L. El Ghaoui 和 M. Arcak，“利用 Christoffel 函数进行数据驱动可达性分析”，载于2021年第60届IEEE决策与控制会议（CDC）。IEEE，2021年，第5067–5072页。[12] W. Dai、B. Pang 和 K. H. Low，“考虑空域占用的城市空中交通无冲突四维路径规划”，《航空航天科学与技术》，文章编号107154，2021年。[13] W. Han、A. Jasour 和 B. Williams，“不确定环境下随机非线性系统的非高斯风险有界轨迹优化”，载于2022年国际机器人与自动化会议（ICRA）。IEEE，2022年，第11044–11050页。[14] M. Chen、S. L. Herbert、M. S. Vashishtha、S. Bansal 和 C. J. Tomlin，“一类非线性系统的可达集与管束分解”，《IEEE自动控制汇刊》，第63卷，第11期，第3675–3688页，2018年。[15] P. Wu 和 J. Chen，“数据驱动不确定性下城市空中交通的在线概率碰撞检测”，载于AIAA SCITECH 2023论坛，2023年，文章编号2539。[16] Y. Zou、H. Zhang、G. Zhong、H. Liu 和 D. Feng，“ 小型无人航空器系统的碰撞概率估计 ”，《可靠性工程与系统安全》，第213卷，文章编号107619，2021年。[17] Y. Yang、J. Zhang、K.-Q. Cai 和 M. Prandini，“基于概率可达集的多航空器冲突检测与解脱”，《IEEE控制系统技术汇刊》，第25卷，第316页，2016年，第1期，第309页。[18] L. Hewing、K. P. Wabersich 和 M. N. Zeilinger，“利用间接反馈实现递归可行随机模型预测控制”，《自动化》，第119卷，文章编号109095，2020年。[19] L. Blackmore、M. Ono、A. Bektassov 和 B. C. Williams，“机会约束随机预测控制的概率粒子控制近似”，《IEEE机器人学汇刊》，第26卷，第517页，2010年，第3期，第502页。[20] H. Zhu 和 J. Alonso-Mora，“动态环境中微型飞行器的机会约束碰撞规避”，《IEEE机器人与自动化快报》，第4卷，第2期，第776–783页，2019年。[21] P. Lathrop、B. Boardman 和 S. Mart´ınez，“分布安全路径规划：Wasserstein安全快速探索随机树”，《IEEE机器人与自动化快报》，第7卷，第1期，第430–437页，2021年。[22] S. Boone 和 J. McMahon，“采用高斯混合的非高斯机会约束航天器机动设计”，载于2022年AAS/AIAA轨道动力学专家会议，2022年。[23] G. S. Aoude、B. D. Luders、J. M. Joseph、N. Roy 和 J. P. How，“通过概率安全运动规划避开具有不确定运动模式的动态障碍物”，《自主机器人》，第35卷，第76页，2013年，第1期，第51页。[24] G. C. Calafiore 和 M. C. Campi，“鲁棒控制设计的情景方法”，《IEEE自动控制汇刊》，第51卷，第5期，第742–753页，2006年。[25] V. Lefkopoulos 和 M. Kamgarpour，“在机会约束轨迹规划中使用不确定性数据”，载于2019年第18届欧洲控制会议（ECC）。IEEE，2019年，第2264–2269页。[26] B. Kehoe、S. Patil、P. Abbeel 和 K. Goldberg，“云机器人与自动化研究综述”，IEEE自动化科学与工程汇刊，第12卷，第2期，第398–409页，2015年。[27] H. Han、S. Fu、H. Sun 和 J. Qiao，“多速率采样数据非线性系统的数据驱动多模型预测控制”，IEEE自动化科学与工程汇刊，2022年。[28] C. Ericson，《实时碰撞检测》。CRC出版社，2004年。[29] M. P. Wand 和 M. C. Jones，《核平滑》。CRC出版社，1994年。[30] K. Ding 和 Q. Zhu，“仅利用部分可测信息的随机时滞切换正系统间歇静态输出反馈控制”，《IEEE自动控制汇刊》，2023年。[31] A. Alanwar、A. Koch、F. Allg¨ower 和 K. H. Johansson，“基于含噪数据的数据驱动可达性分析”，arXiv预印本 arXiv:2105 07229，2021年。[32] B. W. Silverman，《统计学与数据分析中的密度估计》。劳特利奇出版社，2018年。[33] P. Wu 和 J. Chen，“数据驱动不确定性下机会约束地理围栏的在线评估”，载于AIAA AVIATION 2022论坛，2022年，文章编号3613。[34] F. Bullo、J. Cort´es 和 S. Martinez，《机器人网络的分布式控制：运动协调算法的数学方法》。普林斯顿大学出版社，2009年，第27卷。[35] R. Zach，“集合、逻辑、计算：元逻辑开放导论”，2021年。[36] I. Griva、S. G. Nash 和 A. Sofer，《线性与非线性优化》。SIAM

，2009，第 108 卷。

[37] J. E. Mitchell，“用于组合优化问题的分支定界算法”，《应用优化手册》，第 1 卷，2002 年，第 65–77 页。[38] P. S. Efraimidis 和 P. G. Spirakis，“基于蓄水池的加权随机抽样”，《信息处理快报》，第 97 卷，2006 年，第 181–185 页。